LaPS: LiDAR-assisted Placement of Wireless Sensor Networks in Forests

SILVIA DEMETRI, GIAN PIETRO PICCO, and LORENZO BRUZZONE, University of Trento, Italy

The deployment of a wireless sensor network (WSN) is crucial to its reliability and performance. Yet, node placement is typically determined in-field via effort-demanding trial-and-error procedures, because existing approaches over-simplify the radio environment; this especially holds for forests, the focus of this paper, where trees greatly affect communication.

We present LaPS, an approach exploiting remote sensing to identify the best node placement *automatically* and *prior to deployment*. Airborne Light Detection and Ranging (LiDAR) data acquired for the target forest are automatically processed to estimate its properties (e.g., tree position and diameter) that, once incorporated into a specialized path loss model, enable per-link estimates of the radio signal attenuation induced by trees. Finally, a genetic algorithm explores placement options by evolving towards a (sub-)optimal solution while satisfying the user's spatial and network requirements, whose formulation is very flexible and broadly applicable.

Our experiments, focused on a real forest, confirm that LaPS yields topologies of significantly higher quality w.r.t. approaches using a regular placement or a standard path loss model. Further, the ability to quickly explore the impact that changes in user requirements have on topology is invaluable to improve the operation of WSNs and reduce the effort of their in-field deployment.

CCS Concepts: • **General and reference** → *Design*; • **Networks** → **Physical topologies**; Sensor networks;

Additional Key Words and Phrases: Wireless sensor networks, low-power wireless communication, remote sensing, LiDAR, genetic algorithms optimization

ACM Reference Format:
Silvia Demetri, Gian Pietro Picco, and Lorenzo Bruzzone. 2018. LaPS: LiDAR-assisted Placement of Wireless Sensor Networks in Forests. *ACM Trans. Sensor Netw.* 1, 1, Article 1 (January 2018), 39 pages. https://doi.org/10.1145/3293500

1 INTRODUCTION

Wireless sensor networks (WSNs) enable in-situ, unattended monitoring of outdoor natural environments with unprecedented density and flexibility. In the context of forests, which are the main focus of this paper, WSNs have been employed in several applications with different goals, e.g., including forest fire detection, microclimate monitoring, wildlife monitoring [19, 26, 52, 60, 65].

Motivation. In this context, a major challenge is to achieve a proper node placement, as it strongly affects the behavior of communication links [16, 24, 40, 49, 70] and therefore the connectivity of the deployed network and the application performance and reliability at large. However, deploying WSNs in the real world remains a very challenging task, especially in outdoor environments [24, 43,

Authors' address: Silvia Demetri; Gian Pietro Picco; Lorenzo Bruzzone, University of Trento, Department of Information Engineering and Computer Science (DISI), Trento, Italy, firstname.lastname@unitn.it.

1550-4859/2018/1-ART1 $15.00
https://doi.org/10.1145/3293500

63]; forests further exacerbate the challenge due to the presence of trees and vegetation impairing low-power radio communication [13, 46, 68].

In principle, a large literature on network design and node placement optimization exists [2, 8, 30, 34, 53, 69], surveyed in §2. However, the modeling of low-power wireless communication in these approaches lacks realism, as they often neglect the specific features of the real-world target scenario. Since the characteristics of the target environment may vary wildly, they are likely to disrupt the model assumptions, yielding estimates that are unrealistic and of little practical use.

As a consequence, the placement problem is often tackled directly in-field, by means of effort-demanding experimental campaigns. A common approach is to define an initial placement "guess" based on the spatial (e.g., node density) and network (e.g., expected signal strength or number of neighbors) requirements germane to the application, and on the nominal data found in datasheets and derived by idealized radio communication models. Next, the quality of such placement is evaluated in-field by means of connectivity tests, for which several supporting tools exist [14, 29, 62]. However, due to the aforementioned peculiarity of the target environment, it is rarely the case that this initial guess is satisfactory; the position of nodes typically must be nudged based on the outcome of the tests, which must therefore be re-executed, leading to a trial-and-error cycle that repeats until a satisfactory network configuration is found. The effort required by this process obviously increases with the scale of the network and the complexity of the target environment.

In this respect, the forest environment represents both a challenge and an opportunity. The challenge is the fact that, as already pointed out, trees and vegetation impair the radio signal. However, the opportunity is that a significant fraction of this impact is induced by trees, and is therefore *permanent*, i.e., not time-variant, at least not on a short time scale; in other words, the attenuation they induce could be in principle estimated beforehand, and form the basis for determining a satisfactory node placement. This is precisely the goal of this paper.

Approach and contributions. We present an *automatic* node placement approach and companion tool, LaPS (LiDAR-assisted Placement for wireless Sensor networks), that optimize the positions of WSN nodes *prior to deployment* by accounting for the *real characteristics of the target forest.*

The node placement identified by LaPS is subject to a set of simple user-defined spatial and network requirements (§3) specifying desired properties about the placement of nodes and the resulting network connectivity. In this work, the quality of the network layout output by LaPS is assessed in terms of the overall number of communication links and their average expected receive power; however, alternative formulations can be easily encoded, thanks to our flexible design.

A second input to LaPS is information about the forest structure, for which we rely on remote sensing (§4) and specifically airborne Light Detection and Ranging (LiDAR). This technology has been extensively used [5, 27, 33, 44, 50, 61] for accurate estimation of forest attributes (e.g., tree position, trunk diameter, tree density), and LiDAR data is increasingly available. Further, even if the acquisition of airborne LiDAR data is quite expensive, in many cases these data are already acquired for other forest or urban applications, and therefore available at no additional cost.

User requirements and raw LiDAR data are fed to the LaPS toolchain, for which we provide an overview in §5. The first component of the toolchain transforms the raw LiDAR data into a higher-level representation of forest attributes we call a *tree map* (§6). This information provides the crucial parameters of a specialized radio model (§7) that enables accurate per-link estimates of communication quality by taking into account the attenuation induced by trees.

The tree map representing the target forest area and the radio model configured with this information are both input, along with user requirements, to an evolutionary optimization method (§8) that constitutes the last component of the toolchain. Specifically, we exploit genetic algorithms to explore the space of possible placement solutions and evolve towards an optimized placement.

This is done by evaluating the *fitness* of placement configurations in terms of quality and number of the communication links available to network nodes, while honoring the spatial and network requirements set by the user. Genetic algorithms cannot guarantee to identify a global optimum of the fitness function, which is nonetheless very hard to compute given the non-linearity and many dimensions of the problem. In turn, genetic algorithms are of practical relevance as they provide very good sub-optimal solutions with acceptable computational overhead.

To the best of our knowledge, we are the first to exploit LiDAR information with the goal to *i)* characterize[1] the attenuation induced by trees, and *ii)* exploit this and the acquired knowledge about tree positions to identify an efficient placement of WSN nodes.

We validate the LiDAR-based radio model in a real forest (§9), and show that it provides higher accuracy than other coarser-grained approaches found in the literature. Before moving to the evaluation of our placement approach, we discuss the configuration of the LaPS tool (§10) we use for it. This entails not only providing specific examples of user constraint, but also the identification of two parameters, the expected received power and the distance of nodes from trunks, whose minimum value has a relevant impact on the quality of the output solutions and whose value must be determined experimentally.

Finally, we evaluate the quality of the node placements output by LaPS (§11). We analyze various performance metrics, including the overall average expected received power and number of links in the network, along with topological properties quantifying its degree of connectivity and therefore intrinsic robustness. We compare against *i)* a grid-based approach that "blindly" places nodes in each spatial cell without taking into account the presence of trees, and *ii)* a line-of-sight approach that uses the first portion of our LaPS toolchain to determine the links that are unencumbered by trees and for which the standard path loss radio model applies. In essence, the first baseline disregards both the information about the forest structure and the corresponding customized radio model, while the second disregards only the latter. By showing that the placements output by LaPS are significantly better than both baselines we confirm that the additional complexity introduced by LaPS is indeed worthwhile. This becomes even more evident when we introduce a four-fold scale-up of the test network, in which the baselines above cannot even find an acceptable solution.

We conclude our evaluation by quantifying the improvement in quality between the placement solution output by LaPS and the average one. The latter effectively provides a measure of the placement that in-field campaigns are likely to identify. Our results confirm that the improvement attainable by LaPS is significant, thanks to its ability to quickly explore several alternate placements. Therefore, LaPS can provide solutions that *i)* are of higher quality, and *ii)* do not require the significant effort of in-field campaigns. Further, and along the same lines, LaPS enables the exploration of slightly different network and/or performance parameters, to an extent that would be simply prohibitive if performed in-field.

The paper ends in §12 with brief concluding remarks including opportunities for future work.

2 RELATED WORK

Genetic algorithms and WSN node placement. In the last two decades, several studies exploited genetic algorithms towards the planning of mobile networks, in particular for cellular base stations placement and configuration [39, 45, 48]. The effectiveness of these approaches inspired the application of this technique also in the context of WSNs placement optimization.

However, many of the approaches (e.g., [3, 8, 9, 22]) adopt a grid or some other regular pattern (e.g., a tessellation) as the reference network layout, and select candidate positions for node placement only among the vertexes of this regular pattern (e.g., at the crossing of grid lines).

[1]The LiDAR-based radio model was originally published in [17].

This intuitive abstraction simplifies the mathematical treatment of the problem and reduces the search space. However, while it is of practical relevance for some applications (and therefore we consider it hereafter), it is at odds with several other real scenarios, as it neglects specific, and often irregular, placement requirements of applications. In contrast, our approach supports arbitrary spatial requirements where nodes must be deployed in designated areas; the requirements we consider include the common case where nodes must be fastened to trees, yielding an intrinsically irregular structure of the resulting network and search space.

Another significant idealization of the placement problem concerns the communication range, typically assumed perfectly isotropic. In part, this stems from the fact that the majority of WSN approaches aim at jointly optimizing energy consumption and sensing coverage, i.e., minimizing the former without compromising the latter [8, 31, 53]. In this respect, modeling the range of both communication and sensing as a perfect circle is a natural abstraction that, again, greatly simplifies the mathematical treatment. The price to pay, however, is the inability to transfer these approaches in the real world, where communication range is known to be far from isotropic [70].

For instance, the authors of [8] exploit genetic algorithms to optimize a cluster-based approach by determining the best clustering scheme, the operational mode of nodes (e.g., active vs. inactive, slave vs. cluster-head), and their transmission power. However, candidate node positions are restricted to those belonging to a regular grid layout. Communication range is one of the design parameters considered in the optimization; however, it is assumed to depend only on the transmission power, and the impact of the deployment environment is neglected. Similarly, the multi-objective approach presented in [31] optimizes sensor coverage and lifetime, with the additional assumption of a fixed communication and sensing range. Another multi-objective optimization technique is presented in [22] in the context of precision agriculture. Again, the focus is on the selection of node status by taking into account application-specific requirements related to the operation mode, but no effect of the agricultural field is considered in evaluating the connectivity of the network. Analogous considerations hold for the approach in [66], where the optimization of sensing coverage and connectivity is centered around the sleep intervals scheduling for energy conservation while providing different degrees of coverage through their dynamic reconfiguration. However, the connectivity estimation is, once again, unrealistic. The same holds for [53] that, after explicitly mentioning the challenges of real deployments and the difficulty of acquiring prior information about the environment, proceeds to optimize radio coverage and energy consumption based on a circular communication range of fixed radius.

The same idealistic assumption is shared also by the multi-objective optimization of node position and transmission power presented in [34]. This work has an additional point of contact with ours, in that it introduces a k-connectivity constraint on the resulting network, aimed at ensuring some degree of robustness. However, we consider this along with other networking constraints, and analyze their impact on a wider notion of robustness based both on topological properties and considerations common in the design of WSN networking protocols.

Krause et al. [35, 36] also propose a placement approach that simultaneously optimizes communication and sensing quality but, in contrast with the works above, do not rely on the simplistic assumption of fixed communication range. Sensing quality (i.e., the informativeness of sensor positions) and communication cost (i.e., the expected number of retransmissions) are defined by probabilistic models based on Gaussian processes and learned via an initial small pilot deployment. Despite providing strong theoretical guarantees, the approach is not suitable for a forested environment, due to its intrinsic inhomogeneity. The variability in the forest structure strongly affects communication quality, and makes it hard to learn a realistic probability distribution of link quality through a small pilot deployment, which is also very costly to perform in such harsh environment.

In general, LaPS sharply departs from the aforementioned idealized approaches by *i)* taking into account the peculiarity of communication range in WSN with a *specialized* attenuation model, and *ii)* estimating the model parameters directly from the characteristics of the forest area targeted by the deployment, automatically derived via our LiDAR-based toolchain.

Modeling the non-isotropic communication range of WSNs. As already mentioned, the problem of radio range irregularity is well-known. Some works addressed it by adapting the idealized range with a Degree of Irregularity (DOI), defined in [25] as the maximum range variation per unit degree change in the direction of radio propagation. DOI was later refined into the Radio Irregularity Model (RIM) [71], which represents the random variance experimentally observed at each direction via a random weight based on a Weibull distribution.

In contrast, other works attempted to directly model connectivity by describing the statistical behavior of the reception rate. Zuniga et al. [70] identified three regions with different communication quality, depending on link distance: *i)* links in the connected region are stable and symmetric, featuring high reception rate *ii)* links in the disconnected region are almost completely lossy and unreliable *iii)* links in the intermediate transitional region are highly variable and unpredictable. Unfortunately, the transitional region is quite large and includes a significant fraction of the distances typically found in real deployments. Cerpa et al. [15] characterize communication links via non-parametric statistical models that describe the probability density function of the reception rate together with its confidence interval. More precisely, these models relate the reception likelihood with features characterizing both links and groups of links (e.g., distance, asymmetry, uniformity of transmitters and receivers). Again, greater stability is observed in the links belonging to the connected and disconnected regions, which show very high or very low reception rate, respectively. However, although both studies consider different deployment environments (e.g., indoor and outdoor), they do not explicitly take into account their specific characteristics in the analysis. In other words, although these models are derived from experimental evidence, they represent an attempt to generalize a behavior intrinsically determined by the specific deployment scenario.

In this paper, we make a step forward in reconciling the intrinsic generality of modeling and the specificity of the environment at hand, by exploiting the peculiarity of the forest setting and defining a specialized radio propagation model that is the cornerstone of our approach. Although we cannot reproduce every aspect of the target environment affecting communication (e.g., temperature, humidity), we exploit the fact that its quality is severely impacted by the presence of trees, whose position does not change over a short time scale. Therefore, the attenuation introduced by trees can be itself modeled and taken into account when searching for an optimal node placement.

Modeling the signal attenuation induced by vegetation. Over the last decades, several radio propagation models accounting for the presence of vegetation have been presented. The theoretical approach at the base of mechanistics models [58] involves the solution of Maxwell's equations with boundary conditions for each source of scattering along the propagation path. This approach is complex and often not applicable in practice. Therefore, several approximate and simpler empirical models have been developed, notably including the Weissberger [67] and COST 235 [47] models. However, these models assume scenarios where communication links are distant from the ground and traverse the canopy. Consequently, as shown in our evaluation (§9), they are not appropriate for the applications we target, where links are comparatively closer to the ground and traverse the forest at the level of its tree trunks.

Another conventional approach is based on the log-normal path loss model [56], further described in §7. The critical aspect in its application is the estimation of its parameters, whose values are strictly dependent on the specific environment being considered. These parameters are usually determined empirically by regression analysis of in-field measurements. As a consequence, results

are site-specific and suitable only for environments very similar to those where measurements were performed [64]. This approach has been applied for instance in [23], which focuses on the ISM radio bands used by WSNs, and considers propagation paths relatively near to the ground, mainly affected by tree trunks. The authors explicitly consider situations where trees are obstructing the line of sight between transmitters and receivers, deriving distinct models for different obstruction configurations. Nevertheless, these models still lack generality, as their are based on regressions from location-specific measurements.

The work by Azevedo et al. [6] significantly improved the state of the art by explicitly relating the parameters of the log-normal path loss model with a *vegetation index* whose value depends on the average tree density and average trunk diameter in the specific target forest area. This approach is more general, as it clearly identifies the key additional parameters necessary to characterize attenuation in a forest. Nevertheless, the problem then becomes how to determine the vegetation index. In [6], the authors provide values for several type of forests, based on an impressive and effort-demanding in-field campaign. In this paper, we build upon this work and significantly improve it by *i)* providing a methodology, based on LiDAR data (§6), to determine *automatically* (i.e., without in-field campaigns) the vegetation index, and *ii)* extending and refining the attenuation model in [6] to consider the impact of vegetation on a *per-link* basis (§7) instead of across the entire target area, ultimately yielding more accurate estimates (§9).

Exploiting digital models of the environment. The idea of exploiting digital models of a target area and incorporating them in network planning approaches to increase their realism has been applied almost two decades ago by Krzanowsky et al. [37]. Digital Elevation Models (DEMs) and land cover maps were included in a genetic process to tune the expected signal attenuation and compute a more realistic cell coverage, therefore improving the positioning of base stations in wireless networks. More recently, the authors of [49] presented a connectivity model that takes into account topographic and vegetation features, similarly derived from a DEM and land cover maps. However, this model relies on a machine learning algorithm whose training requires the collection of a significant amount of in-field connectivity measures in the target deployment environment.

DEMs are often obtained from satellite or airborne remote sensing systems, whose capabilities have also been explored in this context. In [59] the authors describe how to extract building footprints from LiDAR data with the goal of supporting the design of wireless communications systems in urban areas. These urban features are incorporated in microcell ray tracing models and exploited to assess the visibility status between transmitter and receiver (i.e., visible or obstructed), therefore enabling the selection of a proper radio model (i.e., free space loss vs. single-knife edge diffraction). Similarly, in [38] the line of sight between the satellites and the receiver in a GPS system is evaluated using LiDAR and 3D ray tracing, to assess the positioning accuracy. The work we present here is inspired by these works, in the sense that we similarly exploit the high accuracy of LiDAR to build a detailed model of the environment, and incorporate it into a propagation model. However, these approaches are not directly applicable to our case, as they focus on an urban setting instead of a forest one; further, the peculiarity of the latter, and specifically the need to take into account the attenuation induce by *each* tree, requires a much more fine-grained approach than these works offer.

On the other hand, only few studies in the WSN literature attempted to increase the degree of realism by considering the target scenario and its impact on the effectiveness of node placement. In [2], DEM models are used to estimate the line of sight of PIR sensors in a mountainous region, and determine via an evolutionary approach the placement maximizing sensing coverage. The approach in [30] similarly takes into account coarse-grained elements of the environment (e.g., the presence of vegetation or rivers) known to degrade the sensing capabilities by a given percentage. In [20] an

optimal sensor placement is derived by explicitly accounting for the impact of monsoon seasons and the presence of buildings, with the goal of enabling the prediction of surface wind distribution over a urban water reservoir by relying on a limited number of sensor measurements. However, the impact of the environment on network connectivity is not considered, as the latter relies entirely on GSM—a reliable option in an urban context, but not in a forest. Another attempt to explicitly reckon with the specific environment where the WSN is deployed is presented in [4], which also includes a radio model that *i)* estimates the range on quantized directions around the node by accounting for the presence of obstacles, and *ii)* introduces time-variant environment-dependent components (e.g., a climate factor and an environmental attenuation factor). The results obtained through simulations show that the average range indeed varies based on these factors. However, these are not quantified and the estimation of the corresponding coefficients not discussed; global knowledge about the environmental factors and their impact on communication is instead implicitly assumed.

In contrast, the radio attenuation model at the core of LaPS is validated by in-field experiments; further, to the best of our knowledge, we are the first to exploit remote sensing and specifically LiDAR to derive the fine-grained model of the environment enabling an accurate estimation of attenuation and, ultimately, a significantly better node placement.

3 REQUIREMENTS AND GOALS

We focus on the deployment of WSNs in forests, typically to directly monitor their environment or the animals dwelling in it. Given that the target location is typically harsh and not easy to reach, simplifying the WSN deployment process is crucial, as discussed in §1.

Two factors determine the effectiveness of a WSN deployment in our context. First, *spatial* requirements must be honored, determining how to distribute nodes in the environment to fulfill application needs. Second, a node placement that satisfies only these constraints is not sufficient; *network* requirements must also be fulfilled, to ensure good connectivity among nodes and other properties to be exploited by network protocols towards reliable and efficient communication.

In principle, both spatial and network requirements strongly depend on the application at hand; our approach and associated tools are general enough to accommodate a wide spectrum of requirements. However, in this paper, we focus on common requirements we informally state next.

3.1 Spatial Requirements

Global: REGULAR VS. IRREGULAR. In some applications, e.g., including forest fire detection and microclimate monitoring [19, 24, 26], it is desirable to place nodes in a way that guarantees a spatially uniform sensing with a desired and controlled density. In principle, this could be achieved by organizing nodes in a regular grid, which is indeed a popular choice in placement approaches [69]. In practice, however, an *exact* grid is often not feasible (e.g., due to irregularity of the target environment) or even desirable (e.g., network concerns may suggest a slightly different placement). For these reasons, deployments typically exploit a tessellation of the target area [3, 9, 69] into tiles (e.g., triangles or squares); each node must be contained in one of the tiles. Hereafter, we refer to this deployment scenario as REGULAR.

Other applications are driven by different spatial requirements. For instance, in wildlife monitoring [54, 55] biologists are often interested in monitoring specific sites in the target area (e.g., close to water and feeding sources, or where animal traces are frequently found). Nodes are typically required to be placed near these sites (e.g., within a given radius around a position); in some cases, however, a node may be required to be placed in a designated, geo-referenced position. Moreover, extra relay nodes may be required, whose position is not subject to strict spatial requirements. We refer to this deployment scenario as IRREGULAR.

Node: FREE vs. TRUNK. In addition to the global spatial requirements above, some applications pose also constraints w.r.t. the physical placement of nodes themselves. More precisely, in what we refer to as the FREE placement, nodes can be placed anywhere in the target area, e.g., on the ground or atop poles. However, this option is not viable in some locations, e.g., to avoid that nodes are relocated by weather events or animals, or due to the danger of damaging the tree roots, respectively. Therefore, a common alternative is to fasten the nodes directly to the trees; we refer to this placement as TRUNK. Clearly, these two options implicitly define two different search spaces for node placement, as TRUNK admits acceptable node positions *only* in correspondence of trees.

3.2 Network Requirements

Network requirements ultimately depend on the specific network stack adopted. However, the reliability of the latter is in turn directly affected by the *connectivity* of the network, which we capture by posing requirements on two fundamental metrics on each network node:

- *Minimum expected received power.* This metric effectively defines, in the most basic (and therefore general) way, the minimum acceptable quality of a link. By enforcing this requirement globally, we ensure that all links in the network are "good enough". We further elaborate on this notion in §10.1.
- *Minimum number of neighbors.* This metric builds on the previous one by considering, among all node neighbors, only those with good links. By guaranteeing that a minimum number of these neighbors are available, this metric ensures that each node has enough good communication options—a critical factor for reliability, e.g., in routing protocols.

3.3 Goal: Optimal Node Placement

The goal of our approach is to find an *optimal* node placement, defined informally as a positioning assignment for all nodes of the network that

- satisfies both the spatial and network requirements above, and
- maximizes the quality of the resulting communication network.

Again, several formulations are possible for the latter aspect. Hereafter, we rely on the same basic notions of received power and number of neighbors we exploited in §3.2, and identify the desired placement as the one that maximizes:

- the total number of acceptable links in the network, and
- the average expected received power across all of these acceptable links.

Although these metrics are very basic, when their values are jointly optimized by our approach (§5) they bear a direct effect in shaping the topological characteristics of the network, e.g., how connected is the resulting network graph or how many source-to-sink paths exist in a potential routing topology, as we show in the evaluation (§11).

4 BACKGROUND: REMOTE SENSING AND LIDAR

Remote sensing is an extremely powerful technology to achieve high-precision estimation of tree and forest features [5, 27, 33, 44, 50, 61]. Remote sensing systems measure the electro-magnetic radiation reflected by objects in a scene by sensors mounted, e.g., on satellites, airplanes or UAVs, therefore acquiring data and images over wide areas. By processing these data, many properties of the reflecting objects can be retrieved *automatically*, and therefore with a limited cost in comparison to ground campaigns. Passive optical systems and active LiDAR systems are the remote sensing techniques most commonly used for forest analysis. Optical systems (either multispectral or hyperspectral) measure mostly the horizontal structure of forests, and are therefore suited for the identification of forest areas and for the classification of tree species. In contrast, airborne LiDAR

provides detailed three-dimensional information about the forest structure [27, 33, 44, 50], therefore enabling the accurate estimation of the structural parameters of trees; for this reason, in the rest of the paper we focus on LiDAR only.

Airborne LiDAR systems generate laser pulses that are transmitted from the airplane towards the scene under investigation, in our case a forest area (Figure 1). The pulses hit trees at different heights during their propagation, generating a reflection (called *return*) at every hit that backpropagates towards the pulse source. The sensor, placed at the same location of the source, measures the time of flight between the transmission of the pulse and the reception of its reflected component; this time interval can then be converted into distance. The raw LiDAR data consists of a 3D cloud of measurement points, each containing: *i)* the georeferenced 3D coordinates of the point, *ii)* a return number whose value (ranging from 1 to 4 in our dataset) depends on the time of flight and therefore the level of the forest, and *iii)* a backscattering value, denoting the intensity of the reflected signal. This data enables the 3D reconstruction of the trees in a forest, and therefore the retrieval of information about their position, shape, and size, whose precision is strictly dependent on the spatial density of the emitted laser pulses in the data acquisition phase. High-density (>5 points/m^2) LiDAR data yield the most accurate estimates; Figure 4a shows an example. The estimation of vegetation parameters can be done automatically both w.r.t. individual trees (e.g., height or diameter at breast height) or aggregates (e.g., tree density) depending on the spatial density of available data. In this paper we exploit both, as we further discuss in §6.

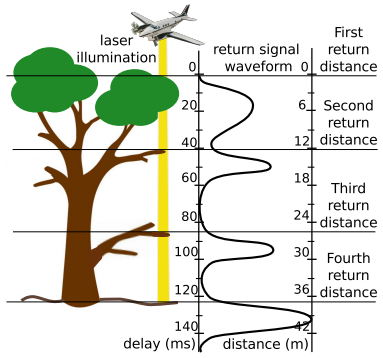


Fig. 1. Airborne LiDAR acquisition mechanism.

5 LIDAR-ASSISTED NODE PLACEMENT: AN OVERVIEW

We now provide the reader with a bird's eye view of the approach we employ to achieve an efficient node placement, for which we exploit the availability of LiDAR information. Figure 2 offers a pictorial representation of the key components and their relationships.

We exploit in combination three main building blocks:

(1) *LiDAR-based forest representation* (§6). It takes as input the geographical representation of the target forest area and the associated LiDAR data and determines a *tree map* encoding, among others, the position of trees and the diameter of their trunks.

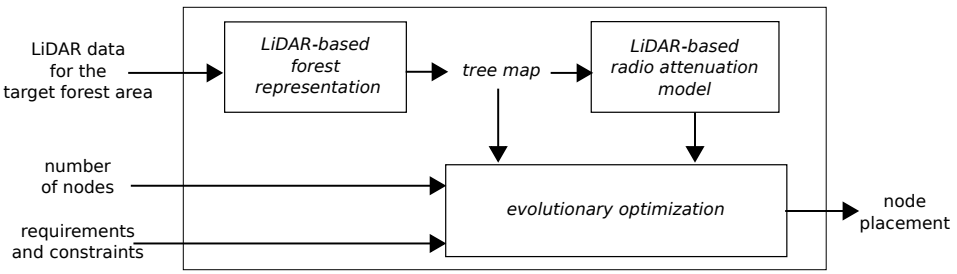


Fig. 2. LiDAR-assisted node placement.

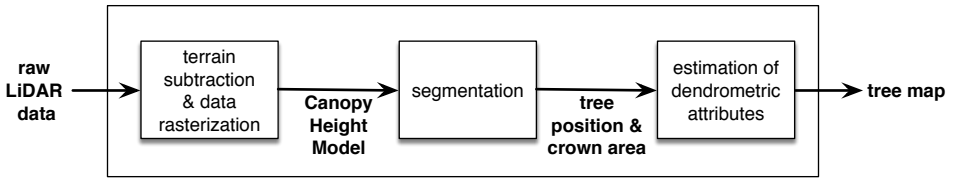


Fig. 3. LiDAR-based forest representation: Data processing chain.

(2) *LiDAR-based radio attenuation model* (§7). It enables *a priori* estimation of the received power of the radio signal based on the aforementioned tree map.
(3) *Evolutionary optimization* (§8). It is the last and most important step, whose inputs are the key parameters of the placement problem, namely, the number of nodes and the spatial and network requirements and constraints (§3). These are combined with the above knowledge about tree positions and their impact on radio propagation and exploited by a genetic algorithm that efficiently explores the search space of feasible placements (i.e., those satisfying spatial and network requirements) and selects the best one.

We next describe in detail each of these building blocks.

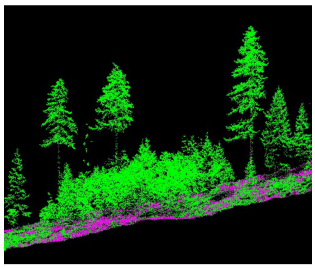
6 LIDAR-BASED FOREST REPRESENTATION

We *automatically* process high-density raw LiDAR data to obtain a *tree map* in which each individual tree in the dominant layer of the forest is represented in terms of its position and other structural attributes, notably including the diameter of its trunk [27, 44]. These information are key to determine the attenuation induced on the radio signal (§7). The data processing chain (Figure 3) takes as input the raw LiDAR data points and performs a sequence of three steps, using a combination of commercial software (e.g., LASTools) and custom tools developed by our research group:

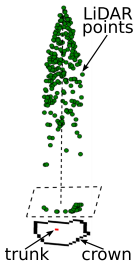
(1) *Extraction of the Canopy Height Model (CHM)*, a raster representation of the height of trees;
(2) *Identification of individual trees*, achieved by properly segmenting the CHM raster image to determine the tree crown;
(3) *Estimation of individual tree attributes*, i.e., tree height, crown radius, and trunk diameter.

Extraction of the Canopy Height Model (CHM). The height of the tree canopy is determined and represented in three stages. First, the terrain morphology is derived from the LiDAR data and mapped into a Digital Elevation Model (DEM) as a discrete surface [5]. The DEM is often available along with the raw LiDAR data; otherwise, it can be derived from the LiDAR 3D point cloud with dedicated tools. The height of the terrain is then subtracted from each raw LiDAR data point to recover its actual elevation from the ground. This elevation, which represents the height of the tree canopy, is then encoded into the CHM with a predefined geometrical resolution (50 cm in our case) that defines the ground area covered by each pixel of the CHM (50×50 cm^2). The encoding assigns to each pixel the maximum height value of the corrected data points belonging to the corresponding ground area.

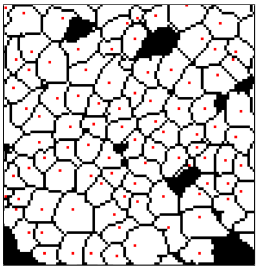
Identification of individual trees. To automatically identify and isolate each tree in the CHM we first identify the tree tops and then delineate each tree crown by segmentation. The former task is achieved by applying a convolutional filter to the CHM raster image to emphasize local maxima; local peaks are then detected via the set level method [33]. The noise generated by grass and other low vegetation is removed by a simple pre-processing; points below a minimum height threshold (2 m in our case) are not considered as belonging to trees. The output of this first phase is a set of



(a) Section of a 3D cloud of points from high-density LiDAR data.



(b) LiDAR points for a single tree and their use in the tree map.



(c) Tree map.

Fig. 4. Representing the forest and tree structure: From raw LiDAR data to the tree map.

pixels representing the tree tops. These constitute the seeds input to the region growing algorithm we use for image segmentation in the second phase, yielding the profile of the tree crowns. This algorithm proceed iteratively, by incrementally expanding the region around each seed with the neighboring pixels until *i)* their canopy height value is below 80% of the seed value, or *ii)* the region diameter exceeds the maximum acceptable value we set at 15 m.

Estimation of individual tree attributes. The two processing steps above yield a tree map in which each tree is represented by *i)* the position of its trunk, corresponding to the position of the tree top, and *ii)* the horizontal projection of the area covered by the tree crown. Figure 4b shows an example of the relation between the raw LiDAR data and these attributes for a single tree; a sample tree map is depicted in Figure 4c.

The last step of the processing in Figure 3 further refines and completes the information in the tree map by enriching it with dendrometric attributes for each tree. The tree height H_L is directly obtained from the CHM value corresponding to the tree top pixel. Further, the tree crown radius K_L is approximated by the radius of the circle whose area is equivalent to the area of the segment representing the crown. Finally, the Diameter at Breast Height (DBH) of each tree trunk can be estimated as a function of H_L and K_L [27]:

$$D = b_0 + b_1 H_L + b_2 K_L + b_3 {H_L}^2 + b_4 {K_L}^2$$

The coefficients $\{b_0, \cdots, b_4\}$ can be estimated via multi-regression by exploiting a small set of ground truth measurements (i.e., tree height, crown radius, and trunk diameter), in our case provided by the local forest service. However, these measurements are not strictly necessary, as alternate approaches exist that do not rely on ground truth. One prominent example are the height-diameter allometric equations widely adopted for forest inventories, which represent the relation between the diameter and the other tree dimensions according to the tree species at hand [44].

7 LIDAR-BASED RADIO ATTENUATION MODEL

We now describe how to exploit the information in the tree map towards a radio attenuation model that enables fine-grained estimation of the expected received signal power in a target forest by accounting for its local characteristics. We assume that communication occurs at the tree trunk level, a common choice in several scenarios of concern [24, 54].

In general, the expected received power P_{rx} [dB] can be estimated, for a given frequency, by

$$P_{rx} = P_{tx} + G_{tx} + G_{rx} - PL$$

where P_{tx} is the transmission power, and G_{tx} and G_{rx} are the receiving and transmitting antenna gains, respectively. In this expression, the last term (i.e., the *path loss*) can be defined by the widely adopted log-normal path loss model [56], which describes the logarithmic decay of the average signal power as a function of the distance d from a transmitter, along with random variations of this received power around the average. In this model, the path loss

$$PL[dB] = PL(d_0) + 10 \cdot n \cdot log\left(\frac{d}{d_0}\right) + X_\sigma$$

depends on three quantities: *i)* $PL(d_0)$, the path loss at a reference distance d_0 in the far field *ii)* n, the path loss exponent defining the attenuation rate w.r.t. distance *iii)* σ, the standard deviation of a zero-mean Gaussian random variable X representing the variation around the average.

The values of these quantities strictly depend on the environment, and are therefore difficult to determine in general. Nevertheless, the authors of [6] observed that, in the context of forests, these parameters of the path loss model are linearly related with local vegetation characteristics, expressed via a *vegetation index* VD that depends on the average density TD [trees/m^2] of trees in the target deployment area and the average diameter D [cm] of their trunks:

$$VD = TD \cdot D \tag{1}$$

For instance, for the 2.4 GHz frequency of interest here, the path loss parameters are:

$$\begin{aligned} PL(d_0) &= -0.82 \cdot VD + 40.1 \\ n &= 0.1717 \cdot VD + 2.2043 \\ \sigma &= 4.4 \end{aligned} \tag{2}$$

The accurate estimation of VD is clearly crucial to this approach. To this end, the authors of [6] do not provide a methodology, rather they rely on data acquired via effort-demanding in-field campaigns. In contrast, the toolchain we illustrated in §6 provides accurate estimates of TD and D in an *automated* fashion, therefore enabling a *practical* and cost-effective application of the specialized path loss model above.

Indeed, the tree map derived from LiDAR data contains the position and diameter of each tree. Therefore, the average density TD is easily obtained by counting the number of trees in the region of interest divided by its area; similarly, D is simply the average of the trunk diameters in the same area. We refer to this direct, yet automated, application of the model in [6] as AREA, as it accounts for the forest structure only in average terms over the entire target area. Unfortunately, the underlying implicit assumption that vegetation and its effect on communication are homogeneous does not hold in practice in real forest environments [40, 64].

For this reason, empowered by the fine-grained information available from our LiDAR-based toolchain, we define a more accurate model that, in contrast, is able to capture the attenuation induced by trees on a *per-link* level; we refer to this second, improved approach as LINK.

The first step towards this goal is the definition of a vegetation index

$$VD_{ij} = TD_{ij} \cdot D_{ij} \tag{3}$$

that is *specific* to each link l_{ij}, instead of being uniform across the target area. In this expression, TD_{ij} and D_{ij} are the density of trees and the average diameter of their trunks in the rectangular area whose length is the line connecting the nodes i and j constituting the link. We refer to this rectangular area as *link area* and set its width to the pixel resolution used to generate the CHM from raw LiDAR data, i.e., 50 cm in this paper. A sample link area is shown in Figure 5.

(a) Clean-LOS. (b) Obstructed-LOS.

Fig. 5. Determining the presence of trees on the line of sight.

The knowledge about the value of VD_{ij} can be used to determine, on a per-link basis, whether communication occurs on a clean line of sight (clean-LOS) or instead is obstructed by the presence of trees (obstructed-LOS), and select accordingly the path loss model that better captures the corresponding situation. Specifically:

(1) if $VD_{ij} = 0$, this means that the link area associated to l_{ij} does not contain any tree trunk, and therefore l_{ij} enjoys a clean line of sight, as in the example of Figure 5a. The corresponding signal attenuation is therefore better estimated by the *free space* path loss model:

$$PL_{free}[dB] = 20log(d) + 20log(f) - 27.55 \tag{4}$$

where d [m] is the distance and f [MHz] the central frequency.

(2) otherwise, if $VD_{ij} \neq 0$, this means that trees are present in the link area, e.g., as shown in Figure 5b. In this case, the behavior of link l_{ij} is better estimated by a "localized" version of Eq. (2), where the average vegetation index VD for the target area is replaced by the link-specific VD_{ij} defined in Eq. (3).

According to these definitions, a link may be obstructed even if no tree is actually sitting on the line connecting the two nodes, as in the case where tree t did not exist in Figure 5b. On the other hand, the resulting processing is very simple, as it consists of checking whether the set of pixels belonging to the link area intersects the set of pixels in which the positions of trunks are mapped.

8 EVOLUTIONARY OPTIMIZATION

As shown in Figure 2, the last component of our approach is the evolutionary optimization, which combines the knowledge derived with the techniques in §6 and §7 with user-defined requirements (§3) about the number and position of nodes, and outputs a (sub-)optimal placement satisfying these requirements in the target forest area.

This best placement cannot be found simply by exhaustive search, which rapidly becomes unfeasible as the number of nodes and/or size of the target area increase. Therefore, we exploit genetic algorithms, a well-known class of numerical optimization procedures inspired by biological evolution that proved effective in optimization problems similar to ours [8, 22, 30, 31, 34, 37].

This class of approaches examines a search space by manipulating and evaluating a set of possible solutions, i.e., a *population* of *individuals*. In our case, an individual is a candidate placement configuration, represented by simply concatenating all node positions; we describe how the latter are determined and in general how the forest area is modeled in §8.1. An individual represents a viable solution only if it satisfies the *constraints* representing the problem at hand; §8.2 provides a formalization of the constraints for our problem, which descend from the spatial and network requirements outlined in §3. Further, an individual is associated with a *fitness* value, a measure of the quality of the individual (and therefore the placement solution it represents) determined by an application-specific function; we describe the one we use in §8.3.

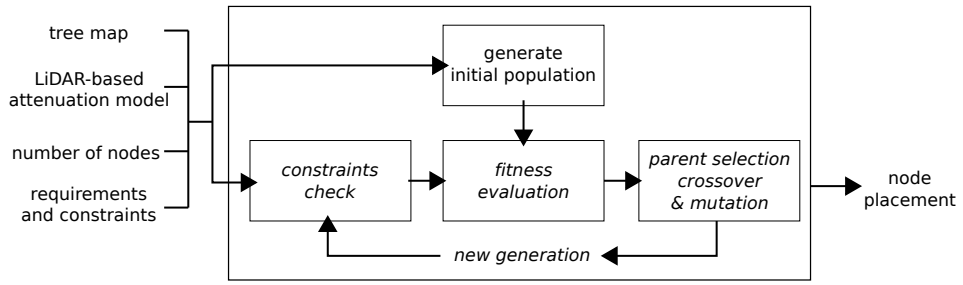


Fig. 6. Evolutionary optimization of placement via genetic algorithms.

The initial population is created by randomly generating a predefined number of individuals, 30 in our current implementation. New populations are created by applying genetic operators to selected individuals, identified by a *parent selection* procedure. The operators are *crossover*, which recombines individuals, and *mutation*, which introduces random variations on individuals based on a predefined probabilistic criterion. Populations evolve iteratively through so-called *generations*; after each iteration, the fitness of each individual is recomputed and becomes the basis for parent selection in the next generation. Eventually, the process converges to a (sub-)optimal solution compliant with the constraints; a global optimum is not guaranteed, but the effect of local optima is mitigated thanks to the random component in the parent selection and application of genetic operators, as described in §8.4.

Figure 6 illustrates our evolutionary optimization based on genetic algorithms.

8.1 System model

The problem is defined starting from the deployment reference scenario (§3.1) and two initial variables: *i)* the node set $\mathbb{N} = \{n_1, \cdots, n_N\}$, with cardinality $N = |\mathbb{N}|$, and *ii)* a target forest area A, for which LiDAR data are available.

By processing raw LiDAR data (§6), we first estimate position and diameter of tree trunks and then represent A by a 2D *tree map* with size $h \times w$. A sample forest area A and the corresponding *tree map* are shown in Figure 7a and 7b, respectively. A uniform quantization is applied to A, based on cells defined by the resolution of the tree map (i.e., 50×50 cm^2). The size of cells represents the spatial granularity we adopt in the analysis, with one quantization cell approximating one possible position where a node can be placed, if constraints are fulfilled.

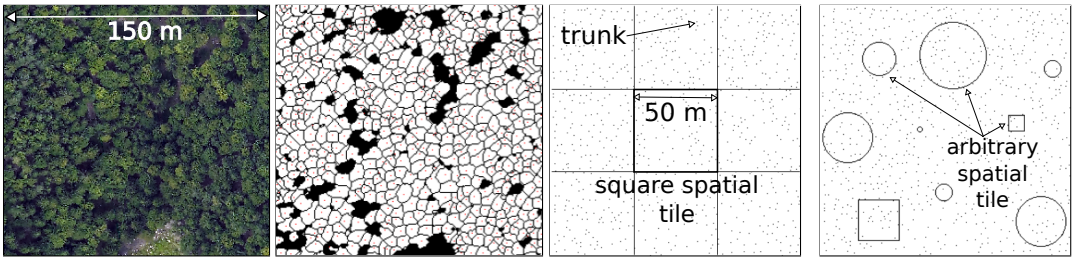
Then, we consider N target spatial *tiles*, i.e., areas where one and only one node can be placed. The size of the tile depends on the reference scenario and on the application requirements (§3.1):

- REGULAR: the target area is divided into N equal-size square tiles. Their size is application-dependent and chosen to cover the whole target area with N nodes. Figure 7c shows an example of 9 tiles covering a sample target area, along with trunk positions derived from the tree map.
- IRREGULAR: the N tiles are sub-areas of A arbitrarily shaped and sized according to the application requirements. An example is reported in Figure 7d.

The design variables are the positions of each network node in the target area. The set of these positions describes the overall node placement, encoded in a 2D scalar vector

$$\mathbb{P} = \{(x_1, y_1), (x_2, y_2), \cdots, (x_i, y_i), \cdots, (x_N, y_N)\}$$

where the position of a node n_i is defined by its scalar coordinates (x_i, y_i) in the target area.



(a) Forest area (b) Tree map (c) REGULAR (d) IRREGULAR

Fig. 7. Division of a sample forest area into 9 spatial tiles: equal-size for REGULAR and arbitrary for IRREGULAR.

A placement $\mathbb{P}$ implicitly identifies also a set of links $\mathbb{L}(\mathbb{P})$. A network link $l_{ij} \in \mathbb{L}(\mathbb{P})$ connects node n_i in position (x_i, y_i) with n_j in (x_j, y_j), $i \neq j$. $\mathbb{L}(\mathbb{P})$ represents all links connecting the nodes in $\mathbb{P}$, and therefore $|\mathbb{L}(\mathbb{P})| = \frac{N(N-1)}{2}$. In practice, however, only a subset of these links are interesting towards our problem, i.e., those whose nodes are in communication range. We denote with $P_{rx}(l_{ij})$ the expected received signal power for link l_{ij}, computed based on the LINK approach (§7) as a function of the distance between n_i and n_j, the presence of trunks on the link line-of-sight, and the transmission power. Since in our radio attenuation model $P_{rx}(l_{ij}) = P_{rx}(l_{ji})$, hereafter we do not distinguish between l_{ij} and l_{ji}; they represent the same link. P_{rx} is at the core at one of the constraints of our placement problem, as described next.

8.2 Constraints

Along the lines of §3, we define separately the spatial and network constraints characterizing the placement problem. Unless otherwise noted, the constraints hold $\forall i, j, k \in \{1, \cdots, N\}$. Interestingly, these constraints are fundamental not only to capture application and system requirements, but also to narrow the search space and consequently reduce the computational overhead.

Spatial constraints. We impose the presence of *exactly one node in each target spatial tile*:

$$A_1: \quad (x_i, y_i) \in tile_i \tag{5}$$

regardless of the shape of the tile (i.e., a square for REGULAR, an arbitrary shape for IRREGULAR).

Apart from this fundamental constraint, others can be defined that further control the placement. For instance, we introduce the following constraints for REGULAR:

$$A_{21}: \quad dist((x_i, y_i), tile_i) \geq \delta_{tile} \tag{6}$$

$$A_{22}: \quad dist((x_i, y_i), (x_j, y_j)) \geq \delta_{link} \tag{7}$$

where *dist* returns the Euclidean distance[2], and δ_{tile} and δ_{link} are threshold values. Essentially, these constraints aim at preserving some degree of uniform coverage: A_{21} avoids *node positions too close to the tile boundary*, and A_{22} imposes a *minimum distance between nodes*.

Similar constraints can be defined for the IRREGULAR scenario, as shown in Figure 7d. However, their precise definition is not particularly interesting, and therefore omitted. In general, alternative and application-dependent constraints are possible. These can be concerned with *sensing* coverage (opposed to the *communication* coverage, the focus of this paper), based on information about the specific characteristics of the sensor like its directionality, as in the case of PIR sensors or cameras. Alternately, they can exploit specifics of the deployment at hand. For example, in the context of a forest fire detection application it may be desirable to place temperature sensors where trees are

[2]For simplicity we assume that *dist* can return also the (minimum) distance of (x_i, y_i) from any of the sides of $tile_i$.

dense and instead measure wind speed in clearings. These constraints can be easily formulated based on the tree map, and automatically accounted for in the companion tool.

Network constraints. We formalize the requirements in §3.2 with the following constraints:

$$C_1 : \quad P_{rx}(l_{ij}) \geq P_{min} \tag{8}$$

$$C_2 : \quad |\mathbb{L}_k| \geq L_{min}, \quad \mathbb{L}_k = \{l_{ij} \in \mathbb{L}_{acc}(\mathbb{P}) \mid i = k\} \tag{9}$$

C_1 sets the *minimum expected received power* P_{min} that defines when a link yields acceptable communication quality; we investigate the appropriate value of P_{min} in §10.1. C_2 defines the *minimum number* L_{min} *of well-connected neighbors* required for each node.

The definition of the latter constraint relies on the set of *acceptable links*, which in principle contains all links satisfying C_1, $\mathbb{L}_{acc}(\mathbb{P}) = \{l_{ij} \in \mathbb{L}(\mathbb{P}) \mid C_1\}$. In practice, however, an additional constraint is needed to set the *minimum distance* δ_{trunk} *of a node from a trunk, along the link line-of-sight*:

$$C_3 : dist((x_i, y_i), (x_t, y_t)) \geq \delta_{trunk} \wedge dist((x_j, y_j), (x_t, y_t)) \geq \delta_{trunk}, \quad \forall (x_t, y_t) \in trees(l_{ij})$$

where $trees(l_{ij})$ returns the position of all tree trunks that intersect link l_{ij}. The set of acceptable links is accordingly defined as

$$\mathbb{L}_{acc}(\mathbb{P}) = \{l_{ij} \in \mathbb{L}(\mathbb{P}) \mid C_1 \wedge C_3\}$$

The additional constraint C_3 is necessary to exclude from the candidate placement situations where tree trunks are on the link line-of-sight and very close to one of the communicating nodes. In these extreme cases, communication quality is significantly degraded; further, this significant degradation is also less accurately captured by the propagation model. We further elaborate on these topics and provide a value for δ_{trunk} in §10.2.

Note how C_3 holds regardless of the node spatial placement, i.e., FREE vs. TRUNK, as it is a condition on the *link* and not on the *node*. Specifically, recall from §3.1 that in a TRUNK scenario nodes are latched on trees, and their distance from the trunk is therefore 0 m. As a consequence, given a node, all links "behind" the trunk it is attached to violate C_3 and are discarded, while all links "in front" are candidates for further evaluation.

8.3 Fitness Function

We now illustrate the definition of the fitness function $F(\mathbb{P})$ that is the basis for evaluating and comparing the candidate placement solutions fulfilling the constraints.

We define $F(\mathbb{P})$ based on two components, which descend from the requirements and goals in §3. The first one is the total number of acceptable links in the network

$$L_{acc} = |\mathbb{L}_{acc}(\mathbb{P})| \tag{10}$$

while the second component is the average expected received power across all acceptable links

$$\overline{P}_{rx} = \sum_{l_{ij} \in \mathbb{L}_{acc}(\mathbb{P})} \frac{P_{rx}(l_{ij})}{|\mathbb{L}_{acc}(\mathbb{P})|} \tag{11}$$

Maximizing the first value increases the chances to build a network that is connected as well as robust, while maximizing the second increases the overall communication quality. As we show in our evaluation (§11.2), their joint optimization yields significantly better placements w.r.t. using either component alone. Therefore, we take both components into account in the fitness function:

$$F(\mathbb{P}) = L_{acc} + \lambda \overline{P}_{rx} \tag{12}$$

Both terms are normalized between 0 and 1, based on the minimum and maximum values attainable. A tuning parameter λ determines which term is predominant and how strongly. A preliminary analysis showed that the best performance is achieved when $\lambda = 1$, yielding a range $[0, 2]$ of variation for the fitness value.

8.4 Parent Selection and Genetic Operators

An individual $(\mathbb{P}, F(\mathbb{P}))$ is represented by the combination of a candidate placement $\mathbb{P}$ and the corresponding value of the fitness function, $F(\mathbb{P})$. New populations are generated by iteratively applying a parent selection procedure and a genetic operator, and recomputing the fitness value.

The *parent selection* procedure selects parent individuals for reproduction, to generate offsprings. We apply the *binary tournament selection* [57] approach, where two individuals are selected at random and their fitness compared; the individual with better fitness is selected as parent. Tournament selection is executed until a pool of parents of predefined size (e.g., half the population) is selected.

The *genetic operator* performs crossover and mutation [57] on the pool of selected parents to produce offsprings, specifically:

- *one-point crossover* randomly selects two different parents, randomly selects a crossover point (i.e., an index in the vector $\mathbb{P}$ of positions associated to an individual), and swaps the node positions beyond this point between the two parents.
- *uniform mutation* applies the following mutation to a single parent chosen at random: each node position in its $\mathbb{P}$ is replaced, with probability $\frac{1}{N}$, with another position in the same spatial tile, uniformly chosen at random among acceptable ones.

We execute them with probability 0.9 and 0.1, respectively; these values favor the exploration of the possible combinations of node positions currently considered as parents before introducing new random ones. The offsprings are then checked against constraints; individuals that do not comply with the constraints are discarded, while the fitness value is recomputed for the others. The latter compliant offsprings are merged with their parents into an *intermediate population*, from which a number (equal to the population size) of individuals with the best fitness is selected, and a new generation is created. Elitism is included, i.e., the current best individual of each generation always survives into the next one.

9 VALIDATION OF THE RADIO SIGNAL ATTENUATION MODEL

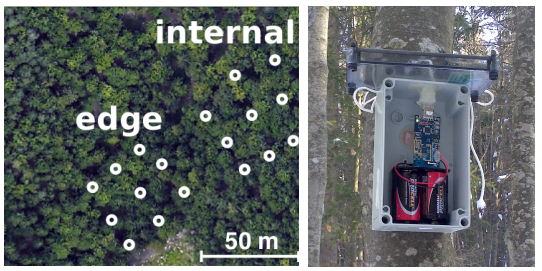
The LiDAR-based radio attenuation model presented in §7 is *validated on real connectivity traces* collected in the field. The validation is based on the comparison between the model predictions and real RSSI measurements collected in two small scale WSN deployments, under these experimental conditions: *i)* absence of snow or rain *ii)* low temperature and humidity *iii)* absence of leaves in the understory vegetation *iv)* almost flat terrain morphology. These conditions implicitly define the assumptions underpinning the evaluation shown here and in §11.

In the following, we describe the forest location used for our study and the setup for RSSI traces collection. We then compare the measurements against the predictions and discuss the results.

9.1 Experimental Setting

Location. The deployment area we are targeting is a typical alpine forest site in Val di Sella (Trentino, Italy), 1000 m above sea level. It is a single-layer forest area (i.e., one single layer of canopy is present) composed of trees belonging to two main species: European beech (*Fagus sylvatica*) and Norway spruce (*Picea Abies*). As this area is a site of interest for the local forest service, we exploit the availability of high-density LiDAR data ($\geq$5 points/m^2) acquired by an Optech ALTM 3100EA sensor mounted on an airborne platform [17].

A WSN composed of 9 nodes is deployed in two different land plots in the target forest, as shown in Figure 8a. The *internal* plot is in the middle of the forest, centered at 46° 0′ 53.64″ N, 11° 22′ 2.51″ E. The *edge* plot is at the fringe of the forest, close to a clearing, with center at 46° 0′ 47.45″ N, 11° 25′ 52.16″ E. Each land plot covers ∼1700 m^2.



(a) (b)

Fig. 8. Experimental setting: (a) location and network topologies and (b) node setup.

Hardware and software. The hardware platform we use is the popular TMote Sky, operating in the ISM 2.4 GHz frequency band according to the IEEE 802.15.4 standards. The radio module is a ChipCon 2420, which includes a digital Direct Sequence Spread Spectrum (DSSS) baseband modem coupled with a digital offset-QPSK modulator, providing an effective data rate of 250 kbps. We use the integrated inverted-F microstrip antenna, which is pseudo-omnidirectional with gain of 3.1 dBi. The software platform supporting the in-field data collection is TRIDENT [29], a tool developed in our group for the untethered execution of communication experiments and collection of connectivity traces. The tool automatically produces the TinyOS code to be installed on the motes, based on the experiment configuration input by the user.

Setup. The same 9-nodes WSN was deployed in a 3×3 grid topology first in the *internal* plot and then in the *edge* plot, as shown in Figure 8a. Nodes were placed into waterproof boxes with the same vertical mounting (i.e., antenna orientation), as shown in Figure 8b. Boxes were then latched to tree trunks at 1.7 m from the ground. To probe the communication link the network nodes broadcast messages in a round-robin fashion: each network node broadcasts in turn one message every 9 s while the other nodes are listening to the radio channel, i.e., one message is sent every second by a different node. When all of the network nodes have sent 200 messages (i.e., after ∼30') one experimental *round* is completed and a new round is started with a different transmission power. More precisely, we alternate transmission power −1 dBm and −8 dBm, which we refer to as *high power* and *low power*, respectively. We use channel 18, i.e., 2.44 GHz with a bandwidth of 3 MHz. Every received message is recorded together with the corresponding *RSSI* value. The noise floor level is also sampled and recorded. The measurement campaign was performed in November 2013. The weather was sunny, with temperature and humidity ranges of $2.5 - 4$℃ and $25 - 45$%, respectively. Overall, we ran 8 experimental rounds and collected 41,794 *RSSI* samples.

The *RSSI* values in our traces are the sum of the received radio signal power and the noise power. Therefore, we can convert those values to a received power indicator, comparable with the predictions of the models, and compute the corresponding P_{rx} by subtracting, in Watt scale, the noise floor level from *RSSI*. This step also makes our analysis independent from hardware differences between nodes w.r.t. noise.

Automatic acquisition of forest parameters. The LiDAR data for the internal and edge forest areas are processed (§6) to obtain the vegetation attributes *TD*, *D* and *VD* characterizing them; the resulting values are reported in Table 1.

We quantitatively evaluate the accuracy of the tree map and of the extracted tree attributes by exploiting the availability of a set of measurements gathered in our internal study site by the local forest service, which performed forest inventory by surveying trees

Table 1. Vegetation parameters in our sites.

site	*D* [cm]	*TD* [trees/m^2]	*VD*
internal	24.2	0.0355	0.8598
edge	25.2	0.0372	0.9366

in sample locations. One of these surveys targets a circular land plot with a 20 m radius, enclosed in our internal site. The forest service mapped 35 trees in this plot, noting their species along with height and trunk diameter at breast height. In comparison, in the same circular plot our processing chain detected 37 trees, of which 32 correspond to an actual one; 91% of trees are therefore correctly detected. More precisely, we obtained 5 wrong detections (or false positives, i.e., tree tops that do not correspond to real trunks) and 3 missed detections (or false negatives, i.e., real tree tops/trunks we did not detect). For the same set of trees, the multi-regression we applied for the estimation of trunk diameters (§6) gives a coefficient of determination $R^2 = 0.7$ and a Root Mean Squared Error $RMSE = 2.75$ cm w.r.t. ground truth values.

These errors generate a difference between the estimated and ground truth values of *TD*, *D* and *VD* of 0.0016 trees/m^2, 0.45 cm, and 0.02, respectively. This affects the estimated received signal power with an error that, at the link distances up to 60 m we consider, is <0.05 dBm. This error is well below the *RSSI* reading accuracy of the radio chip, and therefore can be safely considered negligible. We conclude that our LiDAR toolchain allows us to estimate the relevant vegetation parameters with adequate accuracy.

9.2 Results

AREA model. These forest parameters are used to apply the AREA model and estimate the expected received signal power as a function of link distance, as described in §7.

We quantitatively evaluate the accuracy of the AREA model by comparing its estimated (average) received power $\overline{P}_{rx}$ for the reference distances of our links against data from our in-field traces, by considering separately the low-power and high-power traces as well as the internal and edge sites. Figure 9 shows the curve generated by the AREA model, along with the average *RSSI* measured on links of the same distance in the internal and edge forest areas. We show only the low-power case, as results are similar in the high-power one. The chart confirms that AREA is a good approximation for the real traces. For instance, the average estimation error for the internal site at low power is 4.06 [dBm] with a standard deviation of 3.31 (min. 0.25, max. 12.1). A complete account of estimation errors for both sites and power settings is shown in Table 2.

As a term of comparison, Figure 9 shows also the results from other empirical models often used to estimate the attenuation due to vegetation, i.e., the COST 235 model [47] in its in-leaf (IL) and out-of-leaf (OL) variants, and the Weissberger model [67]. The comparison shows that these models provide significantly worse estimates; this is not surprising, as they are essentially "one-size-fits-all" models assuming homogeneous vegetation, while AREA is based on the *actual* forest structure in our sites, derived *automatically* by the LaPS toolchain. Moreover, the chart shows that these models *underestimate* the actual received power; therefore, if used to guide a WSN deployment, they would likely lead to significant overprovisioning and therefore unnecessary costs.

LINK model. The reasonable accuracy of AREA, obtained by an *aggregate* characterization of the forest area, can be further improved by the *per-link* characterization offered by the LINK model, and its ability to consequently distinguish between clean and obstructed line-of-sight.

Figure 10 reports the P_{rx} measured on the clean-LOS (triangles) and obstructed-LOS links (dots) identified by exploiting the tree map (§7). The received power estimated with the free space path loss model is also depicted, confirming that it better represents the behavior of clean-LOS links, for which the overall average error is reduced from 6.22 dBm to 1.86 dBm at low power, and from 14.21 dBm to 2.71 dBm at high power. Table 2 offers the complete error statistics across models,

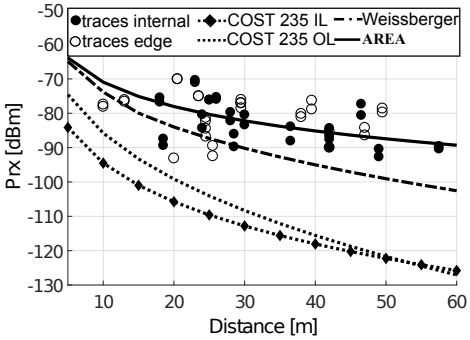


Fig. 9. Received signal power from real traces and various prediction models (low power).

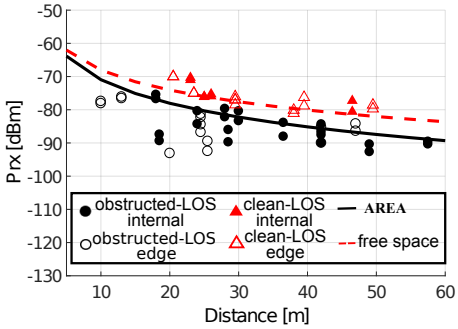


Fig. 10. Comparing traces and estimates in different line-of-sight conditions (low power).

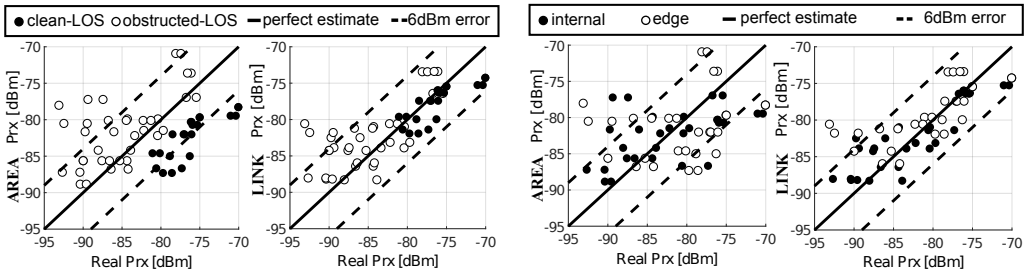
deployment sites, power settings, and line-of-sight situation. By automatically identifying clean-LOS and obstructed-LOS links and applying our diversified strategy accordingly, we significantly and systematically reduce the estimation error for all locations and power settings considered.

Figure 11 provides further evidence by comparing the P_{rx} measured from traces against the one estimated by AREA and LINK. The diagonal of the charts is the perfect estimate with zero error; the dashed lines on its sides mark the ±6 dBm error bands, where this value is the accuracy of the *RSSI* readings provided by the CC2420 radio chip we use. These charts confirm that LINK predictions are more accurate than AREA predictions, for both types of sites and of line-of-sight scenarios.

Figure 11a shows that AREA consistently *underestimates* the received power for clean-LOS points, as these are incorrectly penalized by the aggregate vegetation index *VD*; as a consequence of its removal in LINK, the same points are effectively "shifted" closer to the diagonal, *all* within the ±6 dBm error band. As for the obstructed-LOS, AREA instead *overestimates* the effect of vegetation,

Table 2. Experimental results.

		avg error		std dev		min		max		% in ±6 dBm		% in ±1 dBm	
	site	AREA	LINK	AREA	LINK	AREA	LINK	AREA	LINK	AREA	LINK	AREA	LINK
low power	**internal**	4.06	2.52	3.31	1.99	0.25	0.03	12.10	7.20	76.64	96.67	20.00	33.33
	clean-LOS	6.48	1.97	2.18	2.08	4.21	0.03	9.40	4.94	50.00	100.00	0.00	50.00
	obstructed-LOS	3.18	2.72	3.25	1.97	0.25	0.14	12.1	7.20	86.36	95.54	27.27	27.27
	edge	5.57	3.11	3.32	3.02	0.39	0.40	15.04	12.50	56.67	86.67	6.67	26.67
	clean-LOS	6.07	1.80	1.91	1.48	3.55	0.39	8.77	4.27	57.14	100.00	0.00	35.71
	obstructed-LOS	5.11	4.33	4.27	3.58	0.39	0.39	15.04	12.50	56.25	75.00	12.54	18.75
	all	4.81	2.81	3.38	2.55	0.25	0.03	15.04	12.50	66.67	90.00	13.33	30.00
	clean-LOS	6.22	1.86	1.97	1.68	3.55	0.03	9.40	4.94	54.55	100.00	0.00	40.91
	obstructed-LOS	3.29	3.37	3.76	2.81	0.25	0.14	15.04	12.50	73.68	86.84	21.05	23.68
high power	**internal**	8.20	2.73	4.72	1.69	0.77	0.26	17.58	5.49	36.67	100.00	6.67	13.33
	clean-LOS	14.49	2.94	2.08	2.06	12.33	0.83	17.58	5.49	0.00	100.00	0.00	12.50
	obstructed-LOS	5.91	2.65	2.97	1.58	0.77	0.26	10.19	5.39	50.00	100.00	9.09	13.64
	edge	10.18	4.15	4.89	4.97	0.73	0.43	17.11	14.12	30.00	83.33	3.33	23.33
	clean-LOS	14.06	2.57	1.94	1.66	11.00	0.43	17.11	5.23	0.00	100.00	0.00	28.57
	obstructed-LOS	6.79	5.53	4.08	6.40	0.73	0.42	14.60	14.12	56.25	68.75	6.25	18.75
	all	9.19	3.44	4.87	3.75	0.73	0.26	17.58	14.12	33.33	91.67	5.00	18.33
	clean-LOS	14.21	2.71	1.96	1.77	11.00	0.43	17.58	5.49	0.00	100.00	0.00	22.73
	obstructed-LOS	6.28	3.86	3.46	4.48	0.73	0.26	14.60	14.12	52.63	86.84	7.90	15.79



(a) Line of sight: Clean vs. obstructed.

(b) Deployment site: Internal vs. edge.

Fig. 11. Prediction accuracy of AREA and LINK w.r.t. a ±6 dBm error band.

while LINK generally provides better estimates. The superior accuracy of LINK is confirmed also across the two deployment sites, as shown in Figure 11b. This is confirmed also by the coefficient of determination $R^2 \in [0, 1]$, quantitatively determining how well the estimates represent the measurements, whose value increases from 0.16 for AREA to 0.64 for LINK. As a consequence, the fraction of estimates within a ±1 dBm error significantly increases in LINK (Table 2).

It is interesting to observe that the highest errors (≥6 dBm) in LINK estimates occur in situations where trunks are very close to the nodes; in these cases, LINK tends to overestimate P_{rx}. This aspect is analyzed in more detail in §10.2, via additional experiments in the field.

10 CONFIGURING SPATIAL AND NETWORK CONSTRAINTS

The node placement approach in §5 is based on constraints modeling spatial and system requirements, which are in turn based on configuration parameters whose values we discuss here.

We devote particular attention to constraints C_1 and C_3 in §10.1 and §10.2, as determining their thresholds P_{min} and δ_{trunk} strictly depends on experimental evidence. We offer quantitative considerations based on our own experimental setup that, due to the characteristics of the hardware and environment, is likely to cover a broad spectrum of applications. On the other hand, the methodology we describe can be used to replicate ad hoc examples for different hardware and/or forest environments, enabling one to easily determine the appropriate values for P_{min} and δ_{trunk}.

Finally, in §10.3 we discuss the other parameters that instead depend on generic network and spatial requirements. The values chosen, used in the evaluation (§11), are meant solely to exemplify the flexibility and usefulness of our tool. Table 3 provides a summary of these values in the context of the constraints in which they are used.

10.1 Determining the Minimum Expected Received Power

Constraint C_1 relies on a threshold P_{min} meant to filter out communication links likely to be too unreliable. In ideal conditions, the value of P_{min} is simply the receiver sensitivity threshold. For

Table 3. Configuring application and network constraints.

constraint	parameter	value	definition
A_{21}	δ_{tile}	10 m	minimum distance of a node from the tile boundary
A_{22}	δ_{link}	25 m	minimum link length
C_1	P_{min}	−85 dBm	minimum expected received power at a node
C_2	L_{min}	3	minimum number of acceptable links per node
C_3	δ_{trunk}	5 m	minimum trunk-node distance

example, the CC2420 datasheet specifies a value of −94 dBm; indeed, this was the minimum power level measured for received packets in the experimental campaign described in §10.2.

However, outdoor environments are far from ideal, due to several environmental factors (e.g., temperature and humidity or multipath effects) that affect the radio signal; using the value above is known to lead to unreliable results, as shown by several empirical studies (e.g., [7, 46, 63, 70]). Small signal variations can cause abrupt changes in the ability to receive packets, rendering links unstable. The metric commonly used in these studies is Packet Receipt Ratio (*PRR*), computed as the number of packets received on a link over the number of those sent. It has been shown [70] that when the reception power is close to the receiver sensitivity threshold, links belong to a transitional region with highly variable *PRR*; instead, when the reception power is well above threshold, links belong to a connected region where they exhibit high *PRR*. Hereafter, we aim at ensuring the highest communication quality, therefore retaining only links that belong to the connected region. Other, less conservative choices are easily supported by setting a different value for the threshold P_{min}.

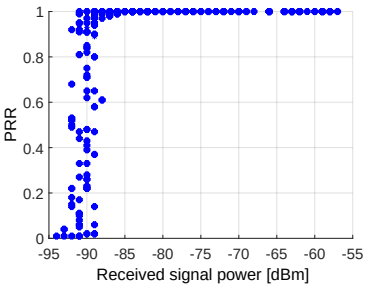


Fig. 12. *PRR* vs average received power.

The relation between *PRR* and received power (*RSSI*) is analyzed in several experimental observations in the literature. In [32], for instance, a good *PRR* is observed for an average $RSSI > -87$ dBm. The measurements we reported in §10.2 confirm these findings. Figure 12 shows the *PRR* as a function of the average received power, computed every 100 packets sent. The charts shows no packet loss for $RSSI \geq -85$ dBm; we therefore select $P_{min} = -85$ dBm as the threshold for C_1.

10.2 Determining the Minimum Trunk-Node Distance

Constraint C_3 specifies the minimum distance δ_{trunk} between the nodes composing a link and the trees on the link line-of-sight. Indeed, when nodes are too close to a tree, the attenuation induced by the latter increases abruptly; further, this increase is not captured accurately by the model in §7.

To determine the threshold value δ_{trunk} beyond which these phenomena occur, we run dedicated in-field experiments in Monte Bondone, near Trento, using TMote Sky motes as in the validation of the LiDAR-based radio attenuation model (§9). The experimental setup is described in Figure 13. We select an isolated trunk and place two nodes at different distances on its opposite sides; the trunk is therefore on the line of sight of the communication link between the nodes. We consider a maximum link length of 60 m, and explore different positions of the trunk inside the link by varying *i)* the distance of the tree from one of the nodes, and *ii)* the link length, i.e., the distance between the two nodes. More precisely, the trunk "moves" w.r.t. one of the nodes in 5 m increments, from 0 to 30 m; at the latter distance, the trunk is in the center of the link. The link length varies from 5 to 60 m; this is achieved in 5 m increments, except for lengths >40 m, for which the increment is 10 m. For each configuration of trunk distance and link length, 100 packets are sent by each node by alternating its packet transmissions (TX) at 1 pkt/s with the other node; overall, one packet is sent on the link every 500 ms. We repeated this process with two TX powers,

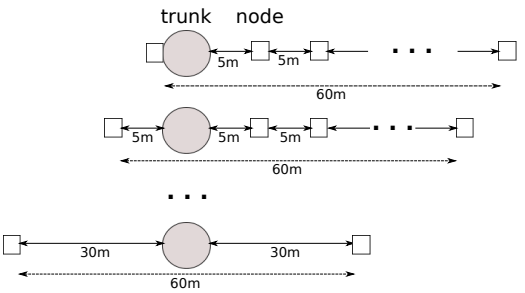
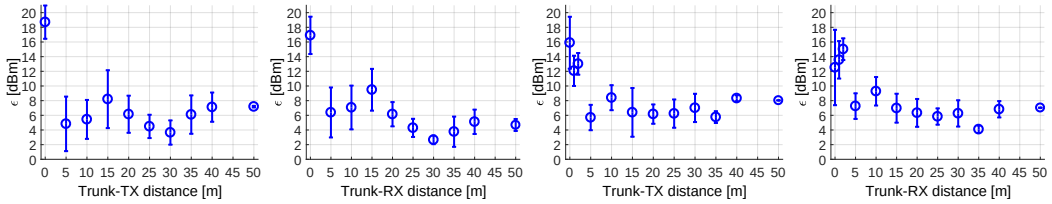


Fig. 13. Determining the minimum trunk-node distance δ_{trunk}: experimental setup.



(a) Trunk diameter: 20 cm. (b) Trunk diameter: 40 cm.

Fig. 14. Average and standard deviation of the error ϵ between the LINK attenuation model in §7 and our measurements when a tree trunk is near to one of the communicating nodes.

−1 and −8 dBm, obtaining similar results; here, we report only those with the higher TX power. Moreover, we repeated experiments with two isolated trees of different diameter, 20 and 40 cm. Overall, we collected ~29600 data points.

We report the average and standard deviation of the error ϵ between our measurements and the estimates of the LINK model in §7, as a function of the distance d between the trunk and the communicating node; further, we separate the cases in which the latter node (which can be on either side of the tree) is transmitting or receiving. ϵ is computed for a given distance d on *all* the link lengths involved; e.g., the value ϵ computed for $d = 5$ m in TX mode is the average of the values for all links in which *either* node is transmitting at 5 m from the trunk. However, each *individual* model estimate depends on the link length; therefore, we compare each link measurement against its corresponding distance-dependent estimate, and obtain the error averaged across all link lengths. The specific diameter at hand is similarly accounted by the model for individual estimates.

Figure 14 shows the results. We observe that the communicating status of a node does not affect significantly the estimation error: whether the trunk is at distance d from a transmitter or a receiver, the effect is approximately the same. On the other hand, the trunk diameter affects the relationship between estimation error ϵ and distance d; however, the error ϵ is overall in line with the results we obtained in §9, except when $d < 5$ m. In this case, our model significantly overestimates the received power by failing to account for the severe attenuation induced by the trunk, regardless of its diameter. The experiments with trunk diameter 40 cm show that this strong attenuation holds not only when the trunk is immediately in front of the communicating node ($d = 0$ m) but also at distance 1 and 2 m; unfortunately, we were not able to gather additional measurements due to logistical reasons.

Based on these considerations, hereafter we conservatively set $\delta_{trunk} = 5$ m in constraint C_3.

10.3 Other Parameters

The other parameters in Table 3 are set as follows. The minimum number L_{min} of acceptable links for a given node (C_2) is key to build a robust network, as it bears a direct impact on the connectivity of the network graph, analyzed quantitatively in §11. We set $L_{min} = 3$, based on the consideration that lower values may easily lead to the creation of several "branches" off the graph. However, in §11.4 we also show that relaxing this constraint may be useful to cope with scale.

As for spatial constraints, we set in A_{21} the minimum distance of a node from the tile border to $\delta_{tile} = 10$ m, and in A_{22} the minimum link length to $\delta_{link} = 25$ m. These values, respectively $\frac{1}{5}$ and $\frac{1}{2}$ of the tile side, are generally reasonable based on our direct experience in real-world deployments. However, here they are mostly meant to be illustrative, as they clearly depend on the application at

hand. Further, as discussed in §8.2, we show the constraint definition only for the case of a REGULAR placement; IRREGULAR ones can be similarly defined.

11 EVALUATING THE QUALITY OF NODE PLACEMENT

We now evaluate the performance of our toolchain, configured as described in §10. We consider spatial constraints stemming from the uniformity of the deployment (REGULAR vs. IRREGULAR) and the mechanics of node positioning (FREE vs. TRUNK), as described in §3. We exploit the same LiDAR dataset acquired for the forest location described in §9, in which we consider the entire 150×150 m^2 area shown in Figure 8a, with center at N 46° 0′ 48.546″, E 11° 25′ 52.122″.

We define the performance metrics and comparison baselines in §11.1, followed in §11.2 by a quantitative analysis of the fitness function $F(\mathbb{P})$ confirming that the linear combination of $\overline{P}_{rx}$ and L_{acc} indeed yields better results w.r.t. the independent use of these components. We then evaluate the performance of our approach in a 9-node deployment inside the aforementioned area (§11.3), followed by a 36-node deployment that allows us to investigate the effect of scaling up the network (§11.4). Finally, we offer a summary of results and a discussion of the main benefits of LaPS (§11.5).

11.1 Performance Metrics and Comparison Baselines

Performance metrics. We analyze quantitatively the networks output by our placement tool based on the same *core metrics* of our optimization approach (§3.3), i.e., the total number L_{acc} of acceptable links and the average expected received power $\overline{P}_{rx}$ across them.

Moreover, we also evaluate the *robustness* of these network from a *topological* standpoint *i)* based on well-known *connectivity* and *centrality* metrics from graph theory, and *ii)* in terms of average number of paths to the sink. The latter is a relevant metric in, e.g., data collection applications based on a tree overlay, as it quantifies the options available to a routing protocol in building (and reconfiguring) routes from each source to the sink.

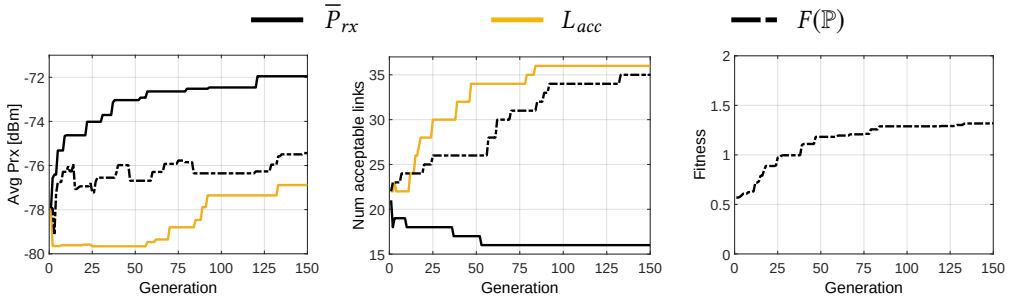
Connectivity is defined as the minimum number of elements (i.e., nodes or links) that, if removed, disconnect the remaining nodes from each other; the higher the connectivity, the more robust the network. We measure this variable by reporting the *average number* $|\overline{\mathbb{L}_k}|$ *of neighbors* with acceptable quality, and computing [18]: *i)* the *binary connectivity* (or *connectedness*) k. Its value is 1 if at least one path exists between all pairs of nodes, 0 otherwise. *ii)* the *vertex connectivity* k_v and *edge connectivity* k_e. They denote, respectively, the minimum number of vertexes and edges that, if removed, disconnect the graph.

An alternative view on robustness is offered by the notion of *betweenness centrality*, defined as

$$b_x = \sum_{i=1}^{N} \sum_{j=i+1}^{N} \frac{s_{ij}(x)}{s_{ij}}$$

where $s_{ij}(x)$ is the number of shortest paths between i and j passing through x and s_{ij} is the total number of shortest paths between i and j [21]. We use the value normalized between 0 and 1, obtained by dividing b_x by the total number $\frac{(N-1)(N-2)}{2}$ of node pairs (x excluded). Betweenness centrality quantifies the extent to which each node lies on the shortest paths between other nodes, and therefore expresses the disruption induced by the removal of such node on the communications among other nodes. We report directly this metric and also exploit it in a force-based representation of the network layout which visually conveys information about the fragility of the network.

Comparison baselines. As mentioned in §2, existing placement approaches for WSNs largely neglect the impact of the environment on communication, let apart taking into account the presence of trees. Therefore, we compare the solutions found by our approach against two closely-related baselines: *i)* a *blind regular* placement where nodes are organized in an exact grid without taking



(a) Average received signal power (b) Number of acceptable links (c) Fitness function

Fig. 15. Evolution of the genetic search for $\overline{P}_{rx}$, L_{acc}, and their linear combination. Curves show the performance metric associated to the best individual in the corresponding generation.

into account the forest characteristics, and *ii)* a *line-of-sight* placement in which no link is obstructed by tree trunks, and therefore the standard path loss model can be applied (§7).

The blind regular placement is a common choice in the literature [8, 9, 22, 69] as it is intuitive and, in principle, of straightforward application in real deployments. Comparing against this placement strategy allows us to *assess whether detailed knowledge of the tree positions is at all useful.*

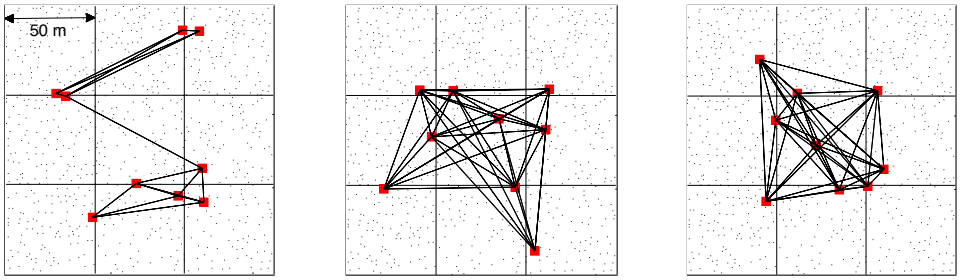
In contrast, the line-of-sight placement exploits precisely this information, derived from LiDAR data via our toolchain, towards a different goal. Indeed, this variant uses alternate constraints w.r.t. those in §8.2, aimed at selecting only those network topologies whose links are unobstructed by trees; in this situation, the specialized radio attenuation model in §7 becomes superfluous, and the standard path loss model can be used instead. Therefore, comparing against this line-of-sight placement allows us to *assess whether a specialized radio model is truly necessary.* Interestingly, however, even in the case of a negative answer the proposed toolchain still constitutes a novel asset, enabling the selection of the topologies, if any, whose links enjoy a free line of sight.

11.2 A Closer Look at the Fitness Function

We begin our evaluation by providing quantitative evidence that our definition of the fitness function $F(\mathbb{P})$ as a linear combination of $\overline{P}_{rx}$ and L_{acc} yields better performance than using either component alone. To better elicit trends, we refer to the scenario REGULAR FREE and neglect the spatial constraints A_{21} and A_{22}. Figure 15 and 16 show the evolution of the genetic search and the resulting network layouts when $F(\mathbb{P})$ is $\overline{P}_{rx}$, L_{acc}, or their linear combination in Eq. 12.

By optimizing only the average reception power we obtain, as expected, a placement with very high overall communication quality ($\overline{P}_{rx} = -71.95$ dBm) in only 121 generations; Figure 15a shows the evolution of the genetic search. However, the price to pay is that the total number of acceptable links is only[3] $L_{acc} = 16$. Figure 15b shows the evolution of L_{acc} during the search; we observe that this value actually decreases as better configurations optimizing $\overline{P}_{rx}$ are found. Moreover, the resulting network layout in Figure 16a shows two node clusters connected by only one link; constraint C_2 on the minimum number of neighbors is satisfied, although by means of a fragile topology prone to partitioning. On the other hand, by optimizing only the number of acceptable links, we more than double their number (L_{acc} =36, Figure 15b) after only 84 generations, and

[3]Remember from §8.1 that links l_{ij} and l_{ji} are not distinguished, and counted as one.



(a) (b) (c)

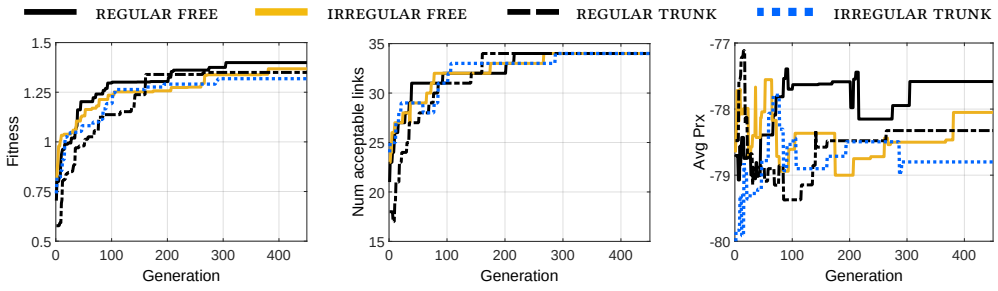
Fig. 16. Network layouts obtained by optimizing (a) $\overline{P}_{rx}$ (b) L_{acc} (c) their linear combination ($\lambda = 1$). Black dots are trunk positions, red squares are node positions, black lines are acceptable links.

Table 4. Core metrics for all scenario combinations and baselines: 9 nodes.

		$F(\mathbb{P})$	L_{acc}	$\overline{P}_{rx}$	$\lvert\overline{\mathbb{L}_k}\rvert$	k	k_v	k_e
LaPS								
REGULAR	FREE	1.39	34	-77.58	7.55	1	6	6
	TRUNK	1.34	34	-78.32	7.55	1	6	6
IRREGULAR	FREE	1.36	34	-78.05	7.55	1	6	6
	TRUNK	1.31	34	-78.79	7.55	1	6	6
blind regular								
REGULAR	FREE	—	17	-78.37	3.77	1	2	2
	TRUNK	—	11	-79.62	2.44	0	0	0
line-of-sight								
REGULAR	FREE	1.29	32	-77.67	7.11	1	6	6
	TRUNK	1.15	28	-76.91	6.22	1	5	5
IRREGULAR	FREE	1.29	31	-76.98	6.88	1	5	5
	TRUNK	1.11	28	-77.57	6.22	1	5	5

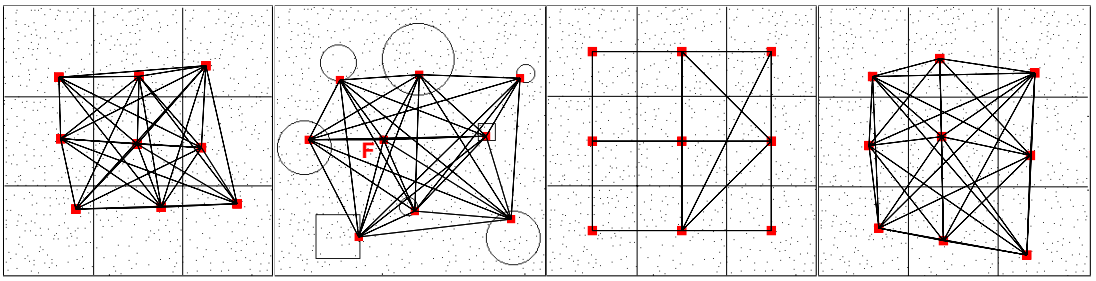
remove the clustering effect from the resulting network topology (Figure 16b). However, this time the price to pay is a significant decrease in the overall link quality ($\overline{P}_{rx} = -76.88$ dBm, Figure 15a).

The definition of $F(\mathbb{P})$ in Eq. 12, which combines $\overline{P}_{rx}$ and L_{acc} with equal weight, strikes a good balance between the two extremes above. Figure 15c reports the value of $F(\mathbb{P}) \in [0, 2]$ (§8.3), monotonically increasing as generations elapse, while Figure 15a and 15b show separately the evolution of its two components. The best solution is found after 136 generations, yielding $L_{acc} = 35$ and $\overline{P}_{rx} = -75.43$ dBm; one link less w.r.t. optimizing only L_{acc}, but with slightly higher overall power. Further, Figure 16c shows that the clustering effect observed when optimizing only $\overline{P}_{rx}$ is absent here, as when optimizing L_{acc}. However, in comparison with Figure 16b, we observe a marked tendency to concentrate the nodes in the center of the target area and place them very close to each other, as this increases the quality of the resulting links. Nevertheless, this undesirable effect is mitigated precisely by the spatial constraints A_{21} and A_{22} we neglected here, but consider in the following sections.



(a) $F(\mathbb{P})$ (b) L_{acc} (c) $\overline{P}_{rx}$

Fig. 17. Evolution of the value of the fitness function and of its components in all scenario combinations.



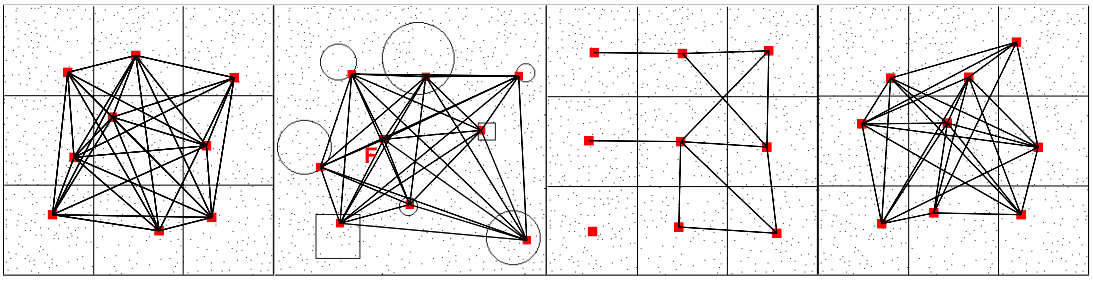
(a) LaPS REGULAR (b) LaPS IRREGULAR (c) blind regular (d) line-of-sight

Fig. 18. Network layouts obtained with FREE node placement.

11.3 Analyzing and Comparing Node Placements

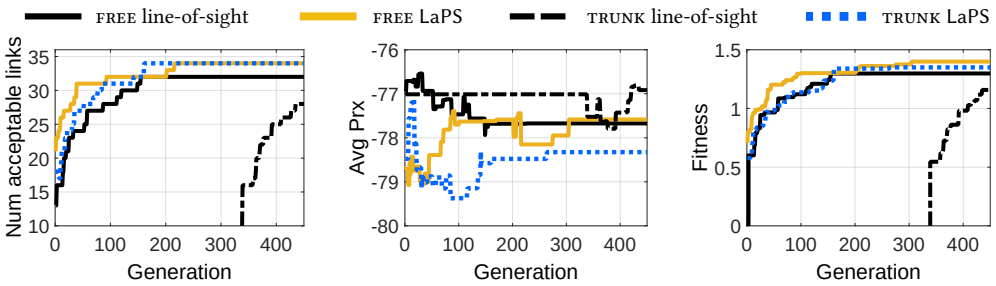
We now analyze the performance of our placement approach when applied to a 9-node network similar to the one we used in §9, and compare it to the blind regular and line-of-sight baselines defined in §11.1. We consider all scenario combinations of REGULAR vs. IRREGULAR and FREE vs. TRUNK placement, subject to all constraints in §3. For IRREGULAR, as shown in Figure 18b we defined a mix of circular and squared regions examplifying spatial tiles required by the application, along with a point F representing a geo-referenced position for which *no* spatial tolerance is allowed.

Core metrics. Table 4 shows the final outcome in terms of fitness value $F(\mathbb{P})$, total number of acceptable links L_{acc}, and average received power $\overline{P}_{rx}$ on these links, while Figure 17 shows the evolution of these values over generations. We observe that the evolution of these metrics is comparable across the various scenarios. The fitness value (Figure 17a) is similar in all cases; the highest value of $F(\mathbb{P}) = 1.39$ is achieved in the case REGULAR FREE, i.e., the least constrained scenario. Similarly, the resulting topologies have the same number of links in the network ($L_{acc} = 34$) and, on average, per node ($|\overline{\mathbb{L}_k}| = 7.55$). This is somewhat surprising considering that the constraints of the four scenarios considered are quite different and, as shown in Figure 18a–18b and Figure 19a–19b, yield network layouts of different shape. On the other hand, the different complexity of the four scenarios is reflected to some extent in the average received power, which is higher in REGULAR



(a) LaPS REGULAR (b) LaPS IRREGULAR (c) blind regular (d) line-of-sight

Fig. 19. Network layouts obtained with TRUNK node placement.



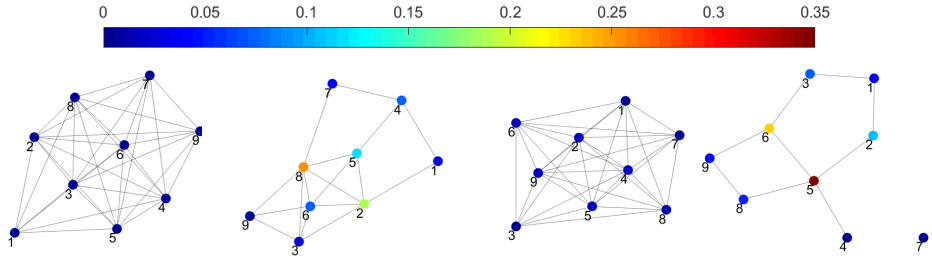
(a) L_{acc} (b) $\overline{P}_{rx}$ (c) $F(\mathbb{P})$

Fig. 20. Evolution of the value of the fitness function and of its components: LaPS vs. line-of-sight in a REGULAR global placement with FREE and TRUNK node placement.

and FREE w.r.t. their IRREGULAR and TRUNK counterparts. Interestingly, the value of $\overline{P}_{rx}$ remains relatively stable across generations (Figure 17c), unlike the value of L_{acc} that increases rapidly (Figure 17b); this is likely due to the constraint A_{22} on the minimum link length distance.

In comparison, the commonly-used and intuitive blind regular placement yields significantly worse results. We obtain this placement by forcing each node exactly in the barycenter of each tile when allowed by a FREE placement or, for a TRUNK placement, next to the tree closest to the barycenter. However, the blind regular placement guarantees the fulfillment of C_2 neither in FREE nor in TRUNK; the value of $F(\mathbb{P})$ cannot therefore be computed. This is shown also in the resulting network layouts of Figure 18c and 19c; in the TRUNK case, one of the nodes is not even connected, and other two are connected via a single link. Finally, the acceptable links are only one half and one third of those obtained by LaPS in FREE and TRUNK, respectively, and with lower $\overline{P}_{rx}$ in both cases.

In contrast, the performance of the line-of-sight placement is significantly better and approaches that of LaPS. The average received power $\overline{P}_{rx}$ is actually higher than LaPS (except for the REGULAR FREE case), since the line-of-sight placement enjoys communication links unobstructed by trees. On the other hand, the number of acceptable links is lower, both for the network as a whole and for individual nodes; the resulting topologies are shown in Figure 18d and 19d. Finally, it is interesting to compare the evolution of fitness for LaPS and line-of-sight (Figure 20). In the FREE case, the values of $F(\mathbb{P})$ are similar although line-of-sight has a slightly worse performance at convergence. However, in the more restrictive TRUNK case, the search problem becomes more complex and the genetic optimization is unable to find valid solutions for more than 300 generations.



(a) LaPS FREE (b) blind regular FREE (c) LaPS TRUNK (d) blind regular TRUNK

Fig. 21. Betweenness centrality. The graphs layout emphasize readability and do not reflect the real topology.

Table 5. Number of paths to the sink for the different placement solutions and scenarios: overall number of paths, overall number of paths with cost ≤ 3 hops and ≤ 2 hops.

		sink	#paths	#paths ≤3 hops	#paths ≤2 hops	sink	#paths	#paths ≤3 hops	#paths ≤2 hops
LaPS									
REGULAR	FREE	central	65890	350	60	upper left	82200	300	48
	TRUNK	central	52516	324	58	upper left	59105	307	53
IRREGULAR	FREE	central	74371	331	55	upper left	76705	321	51
	TRUNK	central	68108	348	60	upper left	69206	332	58
blind regular									
REGULAR	FREE	central	390	59	19	upper left	511	27	8
	TRUNK	central	19	12	8	upper left	23	9	4
line-of-sight									
REGULAR	FREE	central	42078	298	56	upper left	46536	280	50
	TRUNK	central	16320	204	44	upper left	18032	190	38

Connectivity metrics. From Figure 18 and 19 we can visually ascertain that the resulting network topologies are "well connected". We now *quantitatively* assess this aspect, which significantly affects the robustness of the network, by exploiting the graph-based metrics outlined in §11.1, whose values are shown in Table 4.

LaPS always yields connected networks ($k = 1$); to partition them, at least 6 nodes or links ($k_v = k_e = 6$) must fail simultaneously. In contrast, the blind regular placement yields a very vulnerable network in the FREE case, where the loss of only two nodes or links is sufficient to cause a partition; further, the network is not connected in the TRUNK case, as already noted. The line-of-sight placement yields networks that are only marginally less connected than those generated by LaPS. In the REGULAR FREE case the only difference is a slightly lower number of neighbors ($|\overline{\mathbb{L}_k}| = 7.11$), while in the other scenarios only 5 failing nodes or links suffice to partition.

Figure 21 offers an alternative view based on the value of the normalized betweenness centrality. We can observe that this value is similar for all nodes of the networks generated by LaPS, yielding a very robust topology; on the contrary, the blind regular placement yields highly unbalanced topologies in which a handful of nodes are critical, both in terms of reliability (e.g., causing a partition upon crash) and performance (e.g., as potential routing bottlenecks). The corresponding analysis for the line-of-sight placement yields values similar to LaPS, and is therefore omitted.

Table 6. Core metrics for all scenario combinations and baselines: 36 nodes.

		$F(\mathbb{P})$	L_{acc}	$\overline{P}_{rx}$	$\lvert\overline{\mathbb{L}_k}\rvert$	k	k_v	k_e
LaPS								
REGULAR	FREE	0.56	161	-78.04	8.94	1	5	5
	TRUNK	0.54	143	-77.49	7.94	1	4	4
LaPS, $L_{min} = 2$								
REGULAR	TRUNK	0.53	141	-77.48	7.83	1	4	4
line-of-sight								
REGULAR	FREE	—	79	-77.81	4.38	1	1	1
	TRUNK	—	68	-77.51	3.77	0	0	0
line-of-sight, $L_{min} = 2$								
REGULAR	TRUNK	0.35	104	-76.95	5.77	1	2	2

On the other hand, the performance of the two approaches diverges when the network scale is increased, as discussed in §11.4.

Paths to the sink. We conclude our analysis by investigating the effectiveness of placement solutions from a different point of view, closer to the routing layer, and explicitly compare the number of paths to a sink node. Indeed, the more paths available the more options a protocol can exploit to build and reconfigure the routing topology, increasing resilience to failures and link dynamics. For each network layout hitherto examined we compute all possible (acyclic) paths from each node to one identified as a sink. We study two sink positions yielding different path lengths: in the center and in the upper left corner.

Table 5 reports the overall number of possible source-to-sink paths for all combinations of placement modes and scenarios, along with the number of short paths (i.e., up to 2 and 3 hops) particularly desirable for routing purposes. We observe how LaPS yields the highest number of available paths to the sink in all scenarios, both in absolute and w.r.t. short paths. Interestingly, this metric is not directly considered in the genetic evolution; therefore, its high quality is an indirect result of our definition of the fitness function $F(\mathbb{P})$.

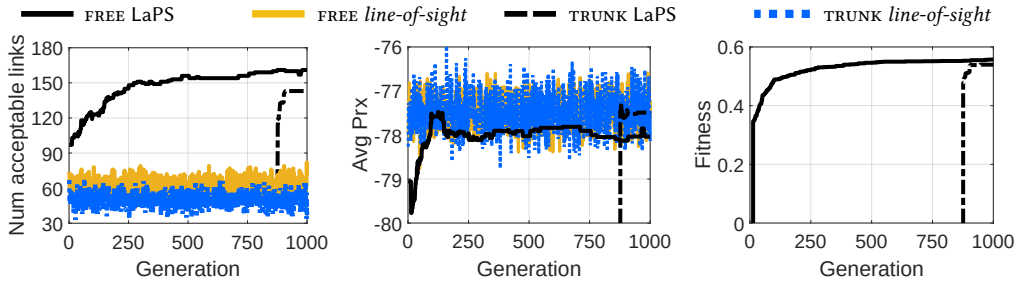
The number of paths does not follow a clear trend across scenarios, as it strongly depends on the specific constraints (e.g., in IRREGULAR) and the particular displacement of tree trunks. The only clear trend is the expected increase in the number of paths when the sink is in the upper left corner.

On the other hand, the aforementioned weakness of the blind regular placement is evident, as it yields two orders of magnitude fewer available paths than LaPS. The line-of-sight placement also consistently generates fewer paths to the sink, as a consequence of its slightly lower connectivity.

11.4 Scaling the Network

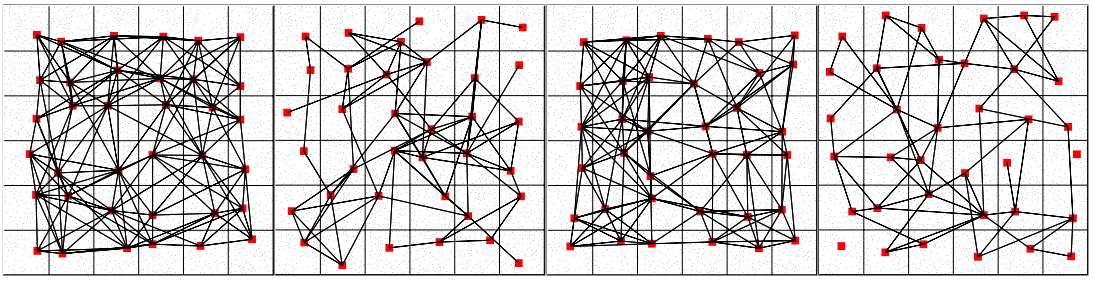
We showed that, even in a small 9-node network, LaPS brings remarkable advantages w.r.t. the commonly-used blind regular placement. This confirms that the knowledge and modeling of the forest structure, acquired via LiDAR data and the LaPS toolchain, plays a key role. On the other hand, the difference with a line-of-sight placement that leverages the same information to *avoid* trees, rather than taking their attenuation into account with our specialized radio model (§7), was less marked. We show next that the difference between these two placement approaches becomes substantial as the size of the target network increases.

Settings. We consider 36 nodes deployed in 300×300 m^2—a *fourfold* increase in both network size and target area. To minimize the bias and enable a direct comparison with the results in §11.3 we simulate the larger area by replicating and spatially recombining the 9 square tiles composing the



(a) L_{acc} (b) $\overline{P}_{rx}$ (c) $F(\mathbb{P})$

Fig. 22. Evolution of the fitness function and of its components for 36 nodes in a 300×300 m^2 forest area, scenario REGULAR.



(a) LaPS FREE (b) line-of-sight FREE (c) LaPS TRUNK (d) line-of-sight TRUNK

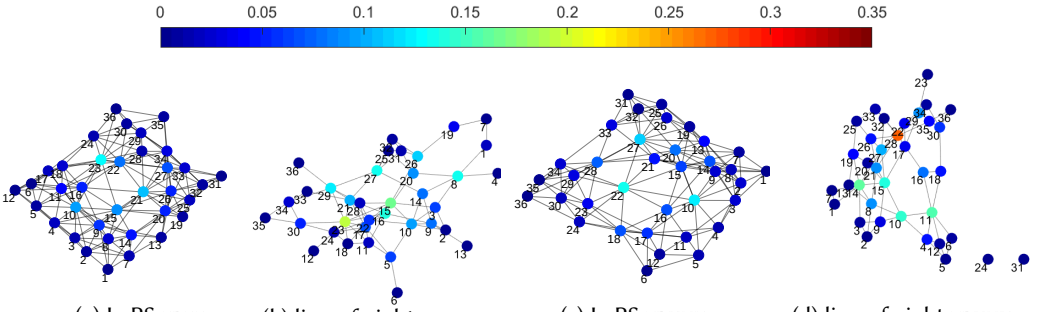
Fig. 23. Network layouts for 36 nodes in a 300×300 m^2 forest area, scenario REGULAR.

smaller target area previously used. The resulting target area (Figure 23) is therefore four times larger than the previous one (Figure 18) but has, by design, the same tree density.

Hereafter, we focus only on the REGULAR scenario but consider both FREE and TRUNK variants. However, unlike §11.3, we do not report about the overall number of paths because, at this scale, an exhaustive search of all possible paths from each node to the sink cannot be performed in polynomial time (#P-complete problem).

FREE node placement. At the scale considered, the placement problem becomes significantly more challenging, even in the less constrained FREE case. Indeed, the line-of-sight approach is unable to find a solution in 1000 generations, as shown in Figure 22; the fitness value cannot be computed, and the genetic search essentially becomes a random search, since no solution fulfilling all constraints is found. This is a consequence of our strict formulation of the genetic search; solutions unable to fulfill the constraints are discarded to avoid misguiding the evolution process.

In contrast, LaPS converges to a solution fulfilling all constraints, characterized by $L_{acc} = 161$ links and an average reception power of $\overline{P}_{rx} = -78.04$ dBm, as shown in Table 6. This demonstrates that, as the scale of the network and/or target area increases, it may become impossible to find a solution in which all links enjoy a free line of sight. LaPS removes this assumption and, thanks to its specialized radio model, can take the tree attenuation into account to identify the best topology.



(a) LaPS FREE (b) line-of-sight FREE (c) LaPS TRUNK (d) line-of-sight TRUNK

Fig. 24. Betweenness centrality for 36 nodes in a 300×300 m^2 forest area, scenario REGULAR.

The latter is shown in Figure 23a, evidencing a balanced structure without clusters. For comparison, Figure 23b reports the network layout corresponding to the line-of-sight placement providing the largest number of links, $L_{acc} = 79$. However, this topology fails to fulfill constraint C_2 about the minimum number of acceptable neighbors.

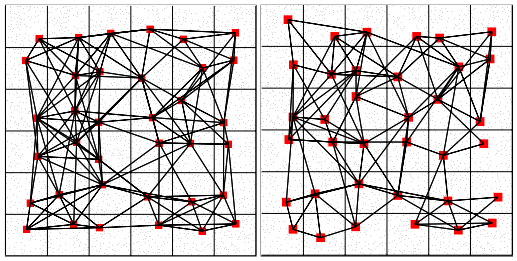
The connectivity metrics in Table 6 also show that the line-of-sight placement, although connected, is quite fragile: a single node or link failure is enough to cause a partition. On the contrary, LaPS yields a rather robust network where $k_v = k_e = 5$ simultaneous failures are required to disconnect the network. This aspect can be better appreciated by comparing Figure 24a and 24b, which visualizes the betweenness centrality using a force-based approach, as in Figure 21. The line-of-sight topology is noticeably less uniform; not only three of the nodes are linked to others via a single connection, but two of the "internal" nodes are significantly more central than others, potentially becoming a traffic bottleneck or a point of failure.

TRUNK node placement. Next, we move to the TRUNK node placement mode, which makes the problem even more challenging at this scale because the overall number of possible node positions is significantly reduced w.r.t. the FREE case.

As shown in Figure 22, the line-of-sight approach is again unable to find acceptable solutions in 1000 generations. In contrast, LaPS begins finding solutions fulfilling all constraints at the 876^{th} generation, and yields a final, high-quality topology as shown in Table 6 and Figure 23c. Although the average power $\overline{P}_{rx}$ achieved is comparable to the line-of-sight case, the number of links in the network (143 vs. 68) and per node (7.94 vs. 3.77) are significantly higher in LaPS. As a result, the network is well-connected ($k_v = k_e = 4$), balanced, and uniform (Figure 24c), while in the line-of-sight case two nodes are isolated and one "internal" node shows a very high betweenness centrality score, leaving the network prone to further partitioning (Figure 24d).

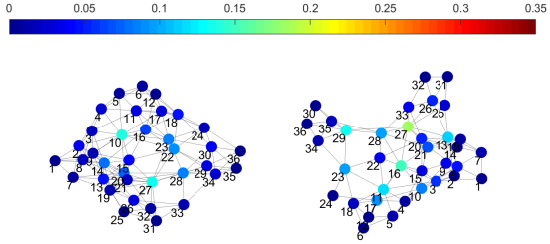
Weakening the constraints. An open question remains as to whether a line-of-sight placement could be achieved with weaker constraints, therefore accepting a compromise between the ability to find a solution and its quality. If a solution can be found, then the related question is what would be the performance of LaPS in the same weakened setting. We investigate these questions next.

In principle, any of the spatial and network constraints we defined in §10 could be relaxed: the ability to quickly explore different tradeoffs is precisely one of the strengths of our approach. To provide an example, we compare the solutions found by LaPS and the line-of-sight approach when constraint C_2 is weakened by reducing the minimum number of acceptable neighbors to $L_{min} = 2$; we focus on the more restrictive TRUNK node placement.



(a) LaPS (b) line-of-sight

Fig. 25. Network layouts for 36 nodes in a 300 × 300 m^2 forest area, scenario REGULAR TRUNK, $L_{min} = 2$.



(a) LaPS (b) line-of-sight

Fig. 26. Betweenness centrality for 36 nodes in a 300 × 300 m^2 forest area, scenario REGULAR TRUNK, $L_{min} = 2$.

With this weakened constraint, the line-of-sight approach converges to a placement solution containing $L_{acc} = 104$ links. The corresponding network is connected, as shown in Figure 25b; however, Figure 26b shows that it is more sparse and fragile than those found by LaPS, both with the weakened constraint ($L_{min} = 2$, Figure 26a) and the original one ($L_{min} = 3$, Figure 24c). Connectivity metrics show that it is sufficient to remove two nodes or links ($k_v = k_e = 2$) to partition the network; further, some nodes appear to be critical to performance and reliability (Figure 26b). With the same configuration, LaPS yields the network layout in Figure 25a with $L_{acc} = 141$ acceptable links (Table 6), only slightly fewer than those obtained with the original constraint $L_{min} = 3$; the average received power and, most important, robust connectivity (Figure 26a) obtained with the two values of L_{min} are similarly comparable.

An interesting observation is that, although the weaker C_2 allows nodes with $L_{min} = 2$ acceptable neighbors, in practice this situation occurs in the line-of-sight case only for 2 nodes (24 and 30); the remaining ones have at least 3 neighbors, and the average $|\overline{\mathbb{L}_k}| = 5.77$ is relatively high. The situation in LaPS is even more stark: *all* nodes have at least 3 neighbors.

This highlights how *i)* the constraints actually specify only a *lower bound* on the desired solutions, that can be overcome during the evolutionary search, and *ii)* the lower is the bound the higher is the number of acceptable topologies explored. In other words, as shown in the case of LaPS, relaxing a constraint may still lead to a very efficient solution, depending on the constraint and specific forest at hand. Once more, the ability to explore several different configurations options *offline* is a remarkable asset of our approach.

11.5 Summary and Discussion

The analysis we presented in this section confirms that LaPS is a powerful tool to understand and examine, *automatically* and *prior to the in-field deployment*, what are the tradeoffs between the application and network requirements and the effectiveness of the placement achievable in the target forest. Further, it also shows that LaPS is able to find very good placements that would be impractical, if not impossible, to find with the trial-and-error approach typically applied in these cases, due to the necessarily limited exploration; the placements yielded by the latter in-field campaigns are therefore more prone to inefficiency and over-provisioning.

However, are these in-field topologies *significantly* worse than the best one output by LaPS? This aspect can be quantified via the fitness function $F(\mathbb{P})$, assuming that the in-field network topologies are contained in the search space. By looking at the range of $F(\mathbb{P})$ values generated during the evolutionary search, we can therefore have a measure of the "distance" from the best case.

	REGULAR		IRREGULAR	
	FREE	TRUNK	FREE	TRUNK
average	0.94	0.94	0.90	0.90
stddev	0.26	0.23	0.22	0.26
best	1.39	1.34	1.36	1.31

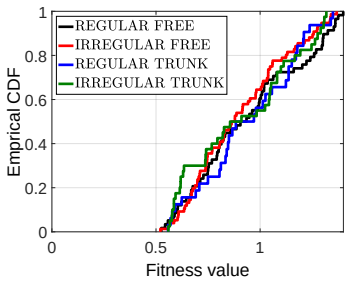


Fig. 27. Statistics and empirical CDF of $F(\mathbb{P})$ values across all scenario combinations (450 generations).

To this end, we focus on the 9-node network analyzed in §11.3, as this represents a rather simple scenario where one would expect that the benefits provided by LaPS are somehow reduced—although our comparison against a common blind regular placement already showed this is not the case. Figure 27 reports, for all acceptable solutions (i.e., fulfilling all constraints) and across all combinations we considered, the maximum (best) value of $F(\mathbb{P})$ corresponding to the final output placement, along with the average and standard deviation; the chart at the bottom also shows the empirical cumulative distribution function (CDF) of $F(\mathbb{P})$ values.

These statistics show that, for all considered scenarios, the final solutions to which LaPS converges are significantly better than the average of all those explored. In other words, the distribution of $F(\mathbb{P})$ values is rather broad, and therefore the probability to select, with an in-field deployment, a severely under-performing placement is quite high; for instance, Figure 27 shows that the probability of obtaining $F(\mathbb{P}) < 1$ is higher than 50% for all scenarios. Table 7 further analyzes the quality of placements for given values of $F(\mathbb{P})$, using the combination REGULAR FREE as an example; it is easy to see that the difference between the best value and the others is significant. Finally, it is worth noting that the $F(\mathbb{P})$ values we report are computed only for network topologies that fulfill all constraints; ensuring that this is the case in an in-field deployment is a rather laborious task in itself, even for the small-scale 9-node network considered.

Table 7. Quality of placement for specific values of $F(\mathbb{P})$ (REGULAR FREE).

$F(\mathbb{P})$	$\overline{P}_{rx}$ (dBm)	L_{acc}	$\lvert\overline{\mathbb{L}_k}\rvert$
0.7 (min)	−78.72	21	4.2
0.94 (median)	−78.63	26	5.6
1.39 (max)	−77.58	34	7.55

Of course, there is a computational overhead associated to LaPS. On a rather low-scale laptop (Apple MacBook 2008, MacOSX 10.7.5) computing a single generation for the 9-node setup we considered takes about 3 minutes, which become 15 minutes for the 36-node one. These figures are the consequence of the many variables at stake, which in turn witness the complexity of the problem. However, we observe that the current implementation can be significantly optimized, as our goal was simply to build a prototype to demonstrate the feasibility and effectiveness of our techniques. In particular, a distributed implementation, amenable for exploiting parallelism on multiple servers in a data center or in the cloud, can be devised by leveraging the vast literature on distributed genetic algorithms [42]. In any case, we observe that the *offline, unsupervised* computing time required by LaPS, even on the scale we reported, is a negligible cost when compared with the human effort currently required by in-field deployments.

Based on these considerations, we argue that the availability of LiDAR data, properly processed in LaPS via the synergistic application of several techniques, constitutes a formidable asset in

effectively and efficiently exploring the space of alternative placement *before* tackling an effort-demanding in-field deployment. Interestingly, finding the best placement is not the only possible use of LaPS, and not necessarily the most interesting or useful. Indeed, LaPS can be exploited also to quickly explore the implications of slight changes in the spatial or network constraints, as we exemplified at the end of §11.4, therefore navigating the space of inherently conflicting tradeoffs. Again, a similar analysis would be prohibitive to perform in-field, and impossible to carry out offline in the absence of precise information about the tree positions and a specialized radio model exploiting it.

12 CONCLUSIONS AND FUTURE WORK

The effective deployment of WSNs in forests is known to be difficult and costly to achieve. Methods of practical applicability, capable of providing realistic placement guidelines while ensuring a connected and efficient network, are essentially lacking. Deployments are often performed in-field via trial and error, a process likely to yield networks with a quality inferior to what potentially achievable, and involving a very high effort.

In contrast, in this paper we presented LaPS, a node placement approach that exploits the increasing availability of LiDAR data in the context of forestry applications to provide an informed node placement layout *automatically* and *prior to the deployment*, accounting for both the actual forest structure and its effect on the network.

LaPS enables off-line automatic exploration and evaluation of placement options in a target forest via three main components: *i)* a representation of the forest derived from LiDAR data, yielding the position and diameter of each tree *ii)* a specialized radio attenuation model exploiting this knowledge to predict the power received on each link *iii)* an optimization strategy based on a genetic evolution process that, along with the other two components, drives the exploration of candidate node placements towards an efficient (sub-)optimal solution. We have shown how, in a real forest scenario, LaPS outperforms alternative placement strategies based on a regular placement or a purely topological free line-of-sight by yielding networks that are significantly more connected and robust. Our results also show that the LiDAR-based forest representation and the specialized radio model are key in achieving this superior performance, especially as the problem size scales, as they drive the search for the best placement with fundamental variables of the target environment. Further, our tool is flexible and open, as it allows great expressiveness in specifying application and network constraints, as well as the desired quality of the solution, encoded in the fitness function. More generally, the ability to quickly explore the impact of changes in the constraints provided by LaPS is an invaluable asset in improving the quality of WSN operation while minimizing the effort of its in-field deployment.

Nevertheless, the work presented here is only an initial, albeit significant, step that can be regarded as a foundation for future research along at least two dimensions. The first dimension concerns the extent to which LaPS is currently capable to model the environment. For instance, we currently consider only the (common) case where network nodes are all at the same height. However, forests can grow on the slope of hills and mountains, and in general on uneven terrain, whose presence on the line of sight among nodes can hinder communication. These aspects could be accounted for by leveraging the accurate LiDAR-based knowledge of the terrain morphology encoded in DEM models (§6).

However, terrain morphology is still a *static* feature of the environment, analogous to the presence of trees we focused on throughout the paper. A more fundamental leap along the same research dimension is instead constituted by the modeling of *dynamic* parameters, e.g., temperature, humidity, or weather conditions like rain and fog. These parameters are known to significantly affect communication, as shown by several experiences and empirical models [10, 11, 46, 68]. In principle,

these findings could be integrated into LaPS since its flexibility in (re)defining the constraints and the fitness function can be exploited to take these dynamic parameters into account when determining a good node placement. However, the intricacies inherent in measuring and dealing with these dynamic quantities, in combination with the complexity of the forest environment, require a dedicated and significant modeling and validation effort. This is well beyond the scope of this paper, which however offers a solid foundation towards this ultimate goal.

The other dimension for future work concerns instead the conceptual contribution put forth in this paper, namely, the observation that the vagaries of low-power wireless communication can be tamed, or at least mitigated, automatically and prior to deployment with the help of remote sensing technology. We showed concretely that this can be achieved for the short-range low-power radios that have been commonplace in the last decade. Nevertheless, new radios are appearing that offer different tradeoffs between range, bandwidth, and energy consumption. An example is LoRa [41], whose sensitivity to the environment in general, and vegetation in particular, has already been observed [12, 28, 51]. The surge of 5G [1] may further exacerbate this problem. These contexts are clearly different from the specific ones we tackled in this paper. However, we argue that the contributions we put forth can be adapted and in general serve as inspiration for techniques that similarly exploit remote sensing to reduce the human effort involved in deploying in-field several low-power wireless devices and, at the same time, improve the expected quality of the resulting placement.

REFERENCES

[1] Mamta Agiwal, Abhishek Roy, and Navrati Saxena. 2016. Next Generation 5G Wireless Networks: A Comprehensive Survey. *IEEE Communications Surveys Tutorials* 18, 3 (2016), 1617–1655.

[2] Vahab Akbarzadeh, Albert Hung-Ren Ko, Christian Gagné, and Marc Parizeau. 2010. Topography-aware sensor deployment optimization with CMA-ES. In *Int. Conf. on Parallel Problem Solving from Nature*. Springer, 141–150.

[3] Fadi Al-Turjman, Hossam S Hassanein, and Mohamed A Ibnkahla. 2009. Connectivity optimization for wireless sensor networks applied to forest monitoring. In *IEEE Int. Conf. on Communications (ICC)*. IEEE, 1–6.

[4] Imen Arfaoui, Ramzi Bellazreg, and Noureddine Boudriga. 2014. A novel realistic irregular radio model to enhance coverage evaluation in wireless sensor networks. In *22nd Int. Conf. on Software, Telecommunications and Computer Networks (SoftCOM)*. IEEE, 160–164.

[5] Peter Axelsson. 2000. DEM generation from laser scanner data using adaptive TIN models. *International Archives of Photogrammetry and Remote Sensing* 33, B4/1; PART 4 (2000), 111–118.

[6] Joaquim AR Azevedo and Filipe E. S. Santos. 2011. An empirical propagation model for forest environments at tree trunk level. *IEEE Trans. on Antennas and Propagation* 59, 6 (2011).

[7] Nouha Baccour, Anis Koubâa, Luca Mottola, Marco Antonio Zúñiga, Habib Youssef, Carlo Alberto Boano, and Mário Alves. 2012. Radio link quality estimation in wireless sensor networks: a survey. *ACM Trans. on Sensor Networks (TOSN)* 8, 4 (2012), 34.

[8] Amol P Bhondekar, Renu Vig, Madan Lal Singla, C Ghanshyam, and Pawan Kapur. 2009. Genetic algorithm based node placement methodology for wireless sensor networks. In *Proc. of the Int. Multiconference of Engineers and Computer scientists*, Vol. 1. Citeseer, 18–20.

[9] Edoardo S Biagioni and Galen Sasaki. 2003. Wireless sensor placement for reliable and efficient data collection. In *Proc. of the 36th Annual Hawaii Int. Conf. on System Sciences*. IEEE, 10–pp.

[10] Carlo Alberto Boano, Hjalmar Wennerstrom, Marco Zuniga, James Brown, Chamath Keppitiyagama, Felix Oppermann, Utz Roedig, Lars-Åke Norden, Thiemo Voigt, and Kay Römer. 2013. Hot Packets: A systematic evaluation of the effect of temperature on low power wireless transceivers. In *In Proc. of ExtremeCom*.

[11] Carlo Alberto Boano, Marco Zúñiga, James Brown, Utz Roedig, Chamath Keppitiyagama, and Kay Römer. 2014. Templab: A testbed infrastructure to study the impact of temperature on wireless sensor networks. In *In Proc. of IPSN*. IEEE, 95–106.

[12] Martin C Bor, Utz Roedig, Thiemo Voigt, and Juan M Alonso. 2016. Do LoRa low-power wide-area networks scale?. In *Proc. of the 19th ACM Int. Conf. on Modeling, Analysis and Simulation of Wireless and Mobile Systems*. ACM, 59–67.

[13] Matteo Ceriotti, Matteo Chini, Amy L Murphy, Gian Pietro Picco, Francesca Cagnacci, and Bryony Tolhurst. 2010. Motes in the jungle: Lessons learned from a short-term WSN deployment in the Ecuador cloud forest. In *Real-World Wireless Sensor Networks (REALWSN)*. Springer, 25–36.

[14] Alberto Cerpa, Naim Busek, and Deborah Estrin. 2003. SCALE: A tool for simple connectivity assessment in lossy environments. *Center for Embedded Network Sensing* (2003).

[15] Alberto Cerpa, Jennifer L Wong, Louane Kuang, Miodrag Potkonjak, and Deborah Estrin. 2005. Statistical model of lossy links in wireless sensor networks. In *Proc. of the 4th Int. Conf. on Information Processing in Sensor Networks (IPSN)*. IEEE, 11.

[16] Yin Chen and Andreas Terzis. 2011. On the implications of the log-normal path loss model: an efficient method to deploy and move sensor motes. In *Proc. of the 9th ACM Conf. on Embedded Networked Sensor Systems (SenSys)*. ACM, 26–39.

[17] Silvia Demetri, Gian Pietro Picco, and Lorenzo Bruzzone. 2015. Estimating Low-power Radio Signal Attenuation in Forests: A LiDAR-based Approach. In *Int. Conf. on Distributed Computing in Sensor Systems (DCOSS)*. IEEE, 71–80.

[18] Reinhard Diestel. 2010. Graph Theory, volume 173 of. *Graduate texts in mathematics* (2010), 5.

[19] David M Doolin and Nicholas Sitar. 2005. Wireless sensors for wildfire monitoring. In *Proc. SPIE*, Vol. 5765. 477–484.

[20] Wan Du, Zikun Xing, Mo Li, Bingsheng He, Lloyd Hock Chye Chua, and Haiyan Miao. 2015. Sensor Placement and Measurement of Wind for Water Quality Studies in Urban Reservoirs. *ACM. Trans. on Sensor Networks (TOSN)* 11, 3 (2015).

[21] Wendy Ellens and Robert E Kooij. 2013. Graph measures and network robustness. *arXiv preprint arXiv:1311.5064* (2013).

[22] Konstantinos P Ferentinos and Theodore A Tsiligiridis. 2007. Adaptive design optimization of wireless sensor networks using genetic algorithms. *Computer Networks* 51, 4 (2007), 1031–1051.

[23] Jose Antonio Gay-Fernandez, Manuel Garcia Sánchez, Inigo Cuinas, Ana Vazquez Alejos, Javier Garcia Sanchez, and Jose Luis Miranda-Sierra. 2010. Propagation analysis and deployment of a wireless sensor network in a forest. *Progress In Electromagnetics Research* 106 (2010), 121–145.

[24] Carl Hartung, Richard Han, Carl Seielstad, and Saxon Holbrook. 2006. FireWxNet: A multi-tiered portable wireless system for monitoring weather conditions in wildland fire environments. In *Proc. of the 4th Int. Conf. on Mobile systems, applications and services (MobiSys)*. ACM, 28–41.

[25] Tian He, Chengdu Huang, Brian M Blum, John A Stankovic, and Tarek Abdelzaher. 2003. Range-free localization schemes for large scale sensor networks. In *Proc. of the 9th Annual Int. Conf. on Mobile computing and networking (MobiCom)*. ACM, 81–95.

[26] Mohamed Hefeeda and Majid Bagheri. 2007. Wireless sensor networks for early detection of forest fires. In *Int. Conf. on Mobile Adhoc and Sensor Systems (MASS)*. IEEE, 1–6.

[27] Marco Heurich. 2008. Automatic recognition and measurement of single trees based on data from airborne laser scanning over the richly structured natural forests of the Bavarian Forest National Park. *Forest Ecology and Management* 255, 7 (2008), 2416–2433.

[28] Oana Iova, Amy L. Murphy, Gian Pietro Picco, Lorenzo Ghiro, Davide Molteni, Federico Ossi, and Francesca Cagnacci. 2017. LoRa from the City to the Mountains: Exploration of Hardware and Environmental Factors. In *Proc. of the Int. Workshop on New Wireless Communication Paradigms for the Internet of Things Workshop (MadCom 2017), co-located with the 14^{th} Int. Conf. on Embedded Wireless Systems and Networks (EWSN)*.

[29] Timofei Istomin et al. 2014. TRIDENT: In-field Connectivity Assessment for Wireless Sensor Networks. *Proc. of ExtremeCom* (2014).

[30] Seo Jae-Hyun, Kim Yong-Hyuk, Ryou Hwang-Bin, Cha Si-Ho, and Jo Minho. 2008. Optimal sensor deployment for wireless surveillance sensor networks by a hybrid steady-state genetic algorithm. *IEICE Trans. on Communications* 91, 11 (2008), 3534–3543.

[31] Damien B Jourdan and Olivier L de Weck. 2004. Layout optimization for a wireless sensor network using a multi-objective genetic algorithm. In *59th Vehicular Technology Conf. (VTC)*, Vol. 5. IEEE, 2466–2470.

[32] Srinivasan Kannan and Philip Levis. 2006. RSSI is under appreciated. In *Proc. of the 3rd Workshop on Embedded Networked Sensors*, Vol. 3031. 239242.

[33] Akira Kato, L Monika Moskal, Peter Schiess, Mark E Swanson, Donna Calhoun, and Werner Stuetzle. 2009. Capturing tree crown formation through implicit surface reconstruction using airborne lidar data. *Remote Sensing of Environment* 113, 6 (2009), 1148–1162.

[34] Andreas Konstantinidis and Kun Yang. 2011. Multi-objective k-connected deployment and power assignment in wsns using a problem-specific constrained evolutionary algorithm based on decomposition. *Computer Communications* 34, 1 (2011), 83–98.

[35] Andreas Krause, Carlos Guestrin, Anupam Gupta, and Jon Kleinberg. 2006. Near-optimal sensor placements: Maximizing information while minimizing communication cost. In *Proc. of IPSN*.

[36] Andreas Krause, Carlos Guestrin, Anupam Gupta, and Jon Kleinberg. 2011. Robust sensor placements at informative and communication-efficient locations. *ACM Trans. on Sensor Networks (TOSN)* 7, 4 (2011), 31.

[37] RM Krzanowski and J Raper. 1999. Hybrid genetic algorithm for transmitter location in wireless networks. *Computers, Environment and Urban Systems* 23, 5 (1999), 359–382.
[38] Jing Li, George Taylor, David Kidner, and Mark Ware. 2008. Prediction and visualization of GPS multipath signals in urban areas using LiDAR Digital Surface Models and building footprints. *Int. J. of Geographical Information Science* 22, 11-12 (2008), 1197–1218.
[39] Kai Lieska, Erkki Laitinen, and Jaakko Lahteenmaki. 1998. Radio coverage optimization with genetic algorithms. In *IEEE 9th Int. Symp. on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Vol. 1. IEEE, 318–322.
[40] Yunhao Liu, Yuan He, Mo Li, Jiliang Wang, Kebin Liu, and Xiangyang Li. 2013. Does wireless sensor network scale? A measurement study on GreenOrbs. *IEEE Trans. on Parallel and Distributed Systems* 24, 10 (2013), 1983–1993.
[41] LoRa Alliance™. 2018. Wide Area Networks for IoT. https://www.lora-alliance.org. (2018).
[42] Gabriel Luque and Enrique Alba. 2011. *Parallel genetic algorithms: theory and real world applications*. Vol. 367. Springer.
[43] Alan Mainwaring, David Culler, Joseph Polastre, Robert Szewczyk, and John Anderson. 2002. Wireless sensor networks for habitat monitoring. In *Proc. of the 1st ACM Int. Workshop on Wireless sensor networks and applications*. ACM, 88–97.
[44] Matti Maltamo, Erik Næsset, and Jari Vauhkonen. 2014. *Forestry applications of airborne laser scanning: concepts and case studies*. Vol. 27. Springer Science & Business Media.
[45] Carsten Maple, Liang Guo, and Jie Zhang. 2004. Parallel genetic algorithms for third generation mobile network planning. In *Int. Conf. on Parallel Computing in Electrical Engineering (PARELEC)*. IEEE, 229–236.
[46] Ramona Marfievici, Amy L. Murphy, Gian Pietro Picco, Federico Ossi, and Francesca Cagnacci. 2013. How Environmental Factors Impact Outdoor Wireless Sensor Networks: A Case Study. In *Proc. of MASS*.
[47] COST 235. 1996. Radio Propagation Effects on Next-generation Fixed-service Terrestrial Telecommunication Systems. Luxemburg. Final Report.
[48] Herve Meunier, E-G Talbi, and Philippe Reininger. 2000. A multiobjective genetic algorithm for radio network optimization. In *Proc. of the Congress on Evolutionary Computation*, Vol. 1. IEEE, 317–324.
[49] Carlos Oroza, Ziran Zhang, Thomas Watteyne, and Steven D Glaser. 2017. A Machine-Learning Based Connectivity Model for Complex Terrain Large-Scale Low-Power Wireless Deployments. *IEEE Trans. on Cognitive Communications and Networking* (2017).
[50] Claudia Paris, Davide Valduga, and Lorenzo Bruzzone. 2016. A hierarchical approach to three-dimensional segmentation of LiDAR data at single-tree level in a multilayered forest. *IEEE Trans. on Geoscience and Remote Sensing* 54, 7 (2016), 4190–4203.
[51] Juha Petajajarvi, Konstantin Mikhaylov, Antti Roivainen, Tuomo Hanninen, and Marko Pettissalo. 2015. On the coverage of LPWANs: range evaluation and channel attenuation model for LoRa technology. In *14th Int. Conf. on ITS Telecommunications (ITST), 2015*. IEEE, 55–59.
[52] Gian Pietro Picco, Davide Molteni, Amy L Murphy, Federico Ossi, Francesca Cagnacci, Michele Corrà, and Sandro Nicoloso. 2015. Geo-referenced proximity detection of wildlife with WildScope: design and characterization. In *Proc. of the 14th Int. Conf. on Information Processing in Sensor Networks (IPSN)*. ACM, 238–249.
[53] Pyari Mohan Pradhan and Ganapati Panda. 2012. Connectivity constrained wireless sensor deployment using multiobjective evolutionary algorithms and fuzzy decision making. *Ad Hoc Networks* 10, 6 (2012), 1134–1145.
[54] Suzanne Prange, Trevor Jordan, Colin Hunter, and Stanley D Gehrt. 2006. New radiocollars for the detection of proximity among individuals. *Wildlife Society Bulletin* 34, 5 (2006), 1333–1344.
[55] Dave Ramsey, Nick Spencer, Peter Caley, Murray Efford, Keith Hansen, Mary Lam, and Des Cooper. 2002. The effects of reducing population density on contact rates between brushtail possums: implications for transmission of bovine tuberculosis. *J. of Applied Ecology* 39, 5 (2002), 806–818.
[56] Theodore S. Rappaport. 1996. *Wireless communications: Principles and Practice*. Vol. 2. Prentice Hall PTR New Jersey.
[57] Colin Reeves. 2003. Genetic algorithms. In *Handbook of metaheuristics*. Springer, 55–82.
[58] Neil C Rogers et al. 2002. A generic model of 1-60 GHz radio propagation through vegetation-final report. *Radio Agency, UK* (2002).
[59] A Roullier-Callaghan, M Al-Nuaimi, and D Kidner. 2003. A topological feature extraction system from LiDAR data with the application of radiowave propagation modelling. In *12th Int. Conf. on Antennas and Propagation (ICAP)*, Vol. 2. IET, 522–525.
[60] Byungrak Son, Yong-sork Her, and Jung-Gyu Kim. 2006. A design and implementation of forest-fires surveillance system based on wireless sensor networks for South Korea mountains. *Int. J. of Computer Science and Network Security (IJCSNS)* 6, 9 (2006), 124–130.
[61] Rebecca A Spriggs, David A Coomes, Trevor A Jones, John P Caspersen, and Mark C Vanderwel. 2017. An Alternative Approach to Using LiDAR Remote Sensing Data to Predict Stem Diameter Distributions across a Temperate Forest Landscape. *Remote Sensing* 9, 9 (2017), 944.
[62] Kannan Srinivasan, Maria A Kazandjieva, Mayank Jain, Edward Kim, and Philip Levis. 2008. SWAT: enabling wireless network measurements. In *Proc. of the 6th ACM Conf. on Embedded Networked Sensor Systems (SenSys)*. ACM, 395–396.

[63] Robert Szewczyk, Joseph Polastre, Alan Mainwaring, and David Culler. 2004. Lessons from a sensor network expedition. In *European Workshop on Wireless Sensor Networks (EWSN)*. Springer, 307–322.

[64] Theodor Tamir. 1967. On radio-wave propagation in forest environments. *IEEE Trans. Antennas Propag.* 15, 6 (1967), 806–817.

[65] Gilman Tolle, Joseph Polastre, Robert Szewczyk, David Culler, Neil Turner, Kevin Tu, Stephen Burgess, Todd Dawson, Phil Buonadonna, David Gay, et al. 2005. A macroscope in the redwoods. In *Proc. of the 3rd Int. Conf. on Embedded Networked Sensor Systems (SenSys)*. ACM, 51–63.

[66] Xiaorui Wang, Guoliang Xing, Yuanfang Zhang, Chenyang Lu, Robert Pless, and Christopher Gill. 2003. Integrated coverage and connectivity configuration in wireless sensor networks. In *Proc. of the 1st Int. Conf. on Embedded Networked Sensor Systems (SenSys)*. ACM, 28–39.

[67] Mark A Weissberger. 1982. *An initial critical summary of models for predicting the attenuation of radio waves by trees*. Technical Report. DTIC Document.

[68] Hjalmar Wennerström, Frederik Hermans, Olof Rensfelt, Christian Rohner, and Lars-Åke Nordén. 2013. A long-term study of correlations between meteorological conditions and 802.15. 4 link performance. In *10th IEEE Int. Conf. on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*. IEEE, 221–229.

[69] Mohamed Younis and Kemal Akkaya. 2008. Strategies and techniques for node placement in wireless sensor networks: A survey. *Ad Hoc Networks* 6, 4 (2008), 621–655.

[70] Marco Zúñiga Zamalloa and Bhaskar Krishnamachari. 2007. An analysis of unreliability and asymmetry in low-power wireless links. *ACM Trans. on Sensor Networks (TOSN)* 3, 2 (2007), 7.

[71] Gang Zhou, Tian He, Sudha Krishnamurthy, and John A Stankovic. 2006. Models and solutions for radio irregularity in wireless sensor networks. *ACM Trans. on Sensor Networks (TOSN)* 2, 2 (2006), 221–262.